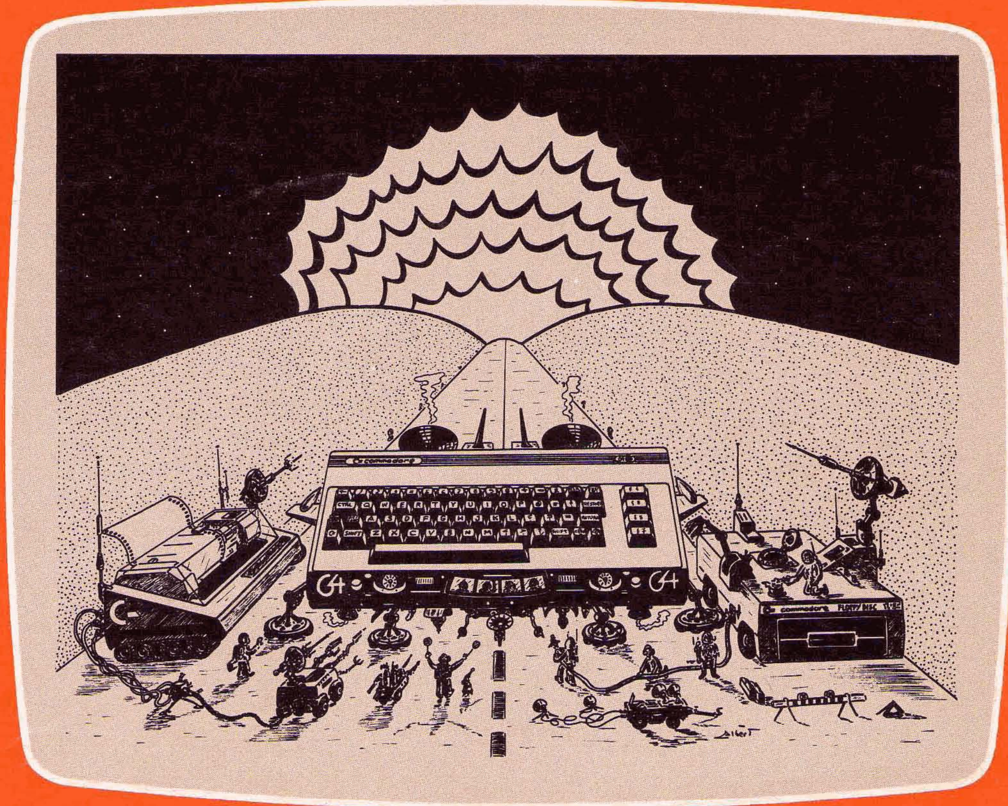
club commodore

Boletín informativo para los usuarios de microordenadores

VIC y CBM



N.º 7

abril 1983

COMMODORE 64 - MAPA DE MEMORIA

Processor I/O Port (6510) Commodore 64

$0000	IN	IN	OUT	IN	OUT	OUT	OUT	OUT	DDR	0
$0001			TAPE MOTOR	TAPE SENSE	TAPE WRITE	D-ROM SWITCH	EFRAM SWITCH	ABRAM SWITCH	PR	1

CIA 1 (IRQ) (6526) Commodore 64

$DC00	PADDLE SEL A B / JOYSTICK O R L D U / KEYBOARD ROW SELECT (INVERTED)	PRA	56320
$DC01	KEYBOARD COLUMN READ	PRB	56321
$DC02	$FF - ALL OUTPUT	DDRA	56322
$DC03	$00 - ALL INPUT	DDRB	56323
$DC04	TIMER A	TAL	56324
$DC05		TAH	56325
$DC06	TIMER B	TBL	56326
$DC07		TBH	56327
~			
$DC0D	TAPE INPUT / TIMER INTERR. B A	ICR	56333
$DC0E	ONE SHOT / OUT MODE / TIME PB6 OUT / TIMER A START	CRA	56334
$DC0F	ONE SHOT / OUT MODE / TIME PB7 OUT / TIMER B START	CRB	56335

CIA 2 (NMI) (6526) Commodore 64

$DD00	SERIAL IN / CLOCK IN / SERIAL OUT / CLOCK OUT / ATN OUT / RS-232 OUT	PRA	56576
$DD01	DSR IN / CTS IN / DCD* IN / RI* IN / DTR OUT / RTS OUT / RS-232 IN — PARALLEL USER PORT	PRB	56577
$DD02	IN / IN / OUT / OUT $3F / OUT / OUT / OUT / OUT	DDRA	56578
$DD03	$06 FOR RS-232	DDRB	56579
$DD04	TIMER A	TAL	56580
$DD05		TAH	56581
$DD06	TIMER B	TBL	56582
$DD07		TBH	56583
~			
$DD0D	RS-232 IN / TIMER B / TIMER A	ICR	56589
$DD0E	TIMER A START	CRA	56590
$DD0F	TIMER B START	CRB	56591

Connected but not used in system

SID (6581) Commodore 64

V1	V2	V3		V1	V2	V3
D400	D407	D40E	FREQUENCY L	54272	54279	54286
D401	D408	D40F	FREQUENCY H	54273	54280	54287
D402	D409	D410	PULSE WIDTH L	54274	54281	54288
D403	D40A	D411	O O O O H	54275	54282	54289
D404	D40B	D412	VOICE TYPE NSE PUL SAW TRI / KEY	54276	54283	54290
D405	D40C	D413	ATTACK TIME 2 ms - 8 sec / DECAY TIME 6 ms - 24 sec	54277	54284	54291
D406	D40D	D414	SUSTAIN LEVEL / RELEASE TIME 6 ms - 24 sec	54278	54285	54292

VOICES (WRITE ONLY)

D415	O O O O O L	54293
D416	FILTER FREQUENCY H	54294
D417	RESONANCE / FILTER VOICES EXT V3 V2 V1	54295
D418	V3 OFF / PASSBAND HI BD LO / MASTER VOLUME	54296

FILTER & VOLUME (WRITE ONLY)

D419	PADDLE X	54297
D41A	PADDLE Y	54298
D41B	NOISE 3 (RANDOM)	54299
D41C	ENVELOPE 3	54300

SENSE (READ ONLY)

SPECIAL VOICE FEATURES (TEST, RING MOD, SYNC) ARE OMITTED FROM THE ABOVE DIAGRAM.

NOTAS TÉCNICAS

En este número iniciamos la publicación del mapa de memoria del COMMODORE 64. Éste presenta la novedad de venir complementado con las cartas individuales de los periféricos, que se publican al principio en esta misma página. En la contraportada interior continúa el mapa de memoria que finalizará en números próximos.

Hex	Decimal	Description
0000	0	Chip directional register
0001	1	Chip I/O; memory & tape control
0003-0004	3-4	Float-Fixed vector
0005-0006	5-6	Fixed-Float vector
0007	7	Search character
0008	8	Scan-quotes flag
0009	9	TAB column save
000A	10	0=LOAD, 1=VERIFY
000B	11	Input buffer pointer/# subscrpt
000C	12	Default DIM flag
000D	13	Type: FF=string, 00=numeric
000E	14	Type: 80=integer, 00=floating point
000F	15	DATA scan/LIST quote/memry flag
0010	16	Subscript/FNx flag
0011	17	0=INPUT;$40=GET;$98=READ
0012	18	ATN sign/Comparison eval flag
0013	19	Current I/O prompt flag
0014-0015	20-21	Integer value
0016	22	Pointer: temporary strg stack
0017-0018	23-24	Last temp string vector
0019-0021	25-33	Stack for temporary strings
0022-0025	34-37	Utility pointer area
0026-002A	38-42	Product area for multiplication
002B-002C	43-44	Pointer: Start-of-Basic
002D-002E	45-46	Pointer: Start-of-Variables
002F-0030	47-48	Pointer: Start-of-Arrays
0031-0032	49-50	Pointer: End-of-Arrays
0033-0034	51-52	Pointer: String-storage(moving down)
0035-0036	53-54	Utility string pointer
0037-0038	55-56	Pointer: Limit-of-memory
0039-003A	57-58	Current Basic line number
003B-003C	59-60	Previous Basic line number
003D-003E	61-62	Pointer: Basic statement for CONT
003F-0040	63-64	Current DATA line number
0041-0042	65-66	Current DATA address
0043-0044	67-68	Input vector
0045-0046	69-70	Current variable name
0047-0048	71-72	Current variable address
0049-004A	73-74	Variable pointer for FOR/NEXT
004B-004C	75-76	Y-save; op-save; Basic pointer save
004D	77	Comparison symbol accumulator
004E-0053	78-83	Misc work area, pointers, etc
0054-0056	84-86	Jump vector for functions
0057-0060	87-96	Misc numeric work area
0061	97	Accum#1: Exponent
0062-0065	98-101	Accum#1: Mantissa
0066	102	Accum#1: Sign
0067	103	Series evaluation constant pointer
0068	104	Accum#1 hi-order (overflow)
0069-006E	105-110	Accum#2: Exponent, etc.
006F	111	Sign comparison, Acc#1 vs #2
0070	112	Accum#1 lo-order (rounding)
0071-0072	113-114	Cassette buff len/Series pointer
0073-008A	115-138	CHRGET subroutine; get Basic char
007A-007B	122-123	Basic pointer (within subrtn)
008B-008F	139-143	RND seed value
0090	144	Status word ST
0091	145	Keyswitch PIA: STOP and RVS flags
0092	146	Timing constant for tape
0093	147	Load=0, Verify=1
0094	148	Serial output: deferred char flag
0095	149	Serial deferred character
0096	150	Tape EOT received
0097	151	Register save
0098	152	How many open files
0099	153	Input device, normally 0
009A	154	Output CMD device, normally 3
009B	155	Tape character parity
009C	156	Byte-received flag
009D	157	Direct=$80/RUN=0 output control
009E	158	Tp Pass 1 error log/char buffer
009F	159	Tp Pass 2 err log corrected



EDITORIAL

¡ya ha llegado el nuevo COMMODORE 64!



Hace pocos días se ha presentado el nuevo modelo de ordenador personal de COMMODORE: el COMMODORE 64. En los últimos tiempos se han recibido numerosas consultas sobre este nuevo equipo que ha despertado una justificada expectación a nivel nacional e internacional. Para información general damos a continuación un resumen de sus características técnicas:

Procesador: *MOS 6510. Posibilidad mediante cartucho de trabajar con el procesador Z-80 y el sistema operativo CP/M.*

Formato de los datos numéricos: *Coma flotante, mantisa de nueve dígitos, exponente de dos dígitos, rango del exponente de −39 a +38.*

Área de memoria del sistema: *20 Kb. de ROM y 3 Kb. de RAM.*

Área del usuario: *38 Kb. de RAM para uso en BASIC o 54 Kb. de RAM si no se usa el intérprete (por ejemplo, en lenguaje máquina u otros lenguajes diferentes de BASIC).*

Teclado: *Tipo QWERTY de 62 teclas con cuatro de función (el teclado es casi lo único igual en el COMMODORE 64 y el VIC-20)*

Pantalla: *40 columnas, 25 filas, 16 colores, salida para televisor standard o monitor. 255 combinaciones pantalla/marco. 256 caracteres incluyendo 124 gráficos por teclado. Hasta ocho bloques de 24×21 puntos definibles por el usuario ("sprites"), cada uno con su propia prioridad en pantalla pudiéndose mover independientemente por toda ella. Además se puede detectar la colisión entre los diferentes bloques y cualquier otra información presente en la pantalla.*

Gráficos en alta resolución: *320×200 puntos.*

Sonido: *Tres voces de nueve octavas cada una con cuatro formas de onda: diente de sierra, triangular, cuadrada con ancho de pulso variable y ruido. Generadores programables independientes para cada voz de ataque, decaimiento, sostenimiento y relajación (ADSR). Filtro programable: paso bajo, paso alto, pasa banda y banda rechazada; frecuencia y resonancia programables. Entrada para señal exterior en el filtro. Control general de volumen.*

Entradas/Salidas: *"Port" de ocho bits paralelo más dos de sincronización. Enlace serie RS-232C implementado en Hardware y Software. Sólo necesita un cartucho de conversión de nivel para ser completamente operativo. Enlace serie para discos e impresora. Bus de expansión para la conexión de cartuchos con programas en ROM. Dos conexiones para "Joystick", "paddles" o lápiz óptico.*

Lenguaje: *BASIC V2 de COMMODORE residente. Otros lenguajes en opción.*

Como puede verse el nuevo COMMODORE 64 es todo un equipo. Sus posibilidades le hacen el más versátil de los ordenadores personales actualmente en el mercado. Como información complementaria diremos que el panorama de Software para el C-64 se está ampliando considerablemente a una velocidad vertiginosa. Seguiremos informando. ■

VENTANA CBM

lectura de "strings" (cadenas) de longitud fija

por **JOSEP TINTO**

El método usual para leer variables provenientes de un fichero de disco es mediante las instrucciones GET# e INPUT#. Sin embargo, las instrucciones GET# e INPUT# indistintamente tienen algunos inconvenientes en determinados casos que seguidamente enumero:

INSTRUCCIÓN INPUT #

La instrucción INPUT# tiene las siguientes limitaciones:

a) Longitud máxima de variable a leer de 79 + CR.

b) Detención de la lectura en una

(continua en la pág. siguiente)

TRUCOS **teclas de función**

por **P. MASATS**



Entre nuestras notas entresacamos hoy ésta que permite disponer de hasta 32 teclas de función diferentes en el VIC. Se consiguen con las cuatro teclas de función, en combinación con las diferentes teclas especiales (son las teclas SHIFT, COMMODORE y CTRL); estas teclas al pulsarse en diferentes combinaciones dan los siguientes resultados en la posición de memoria 653:

TECLA PULSADA	VALOR DE 653
Ninguna tecla	0
SHIFT	1
COMMODORE	2
SHIFT+COMMODORE	3
CTRL	4
SHIFT+CTRL	5
COMMODORE+CTRL	6
SHIFT+COMM+CTRL	7

Las propias teclas de función proporcionan los siguientes resultados en la posición de memoria 203:

TECLA PULSADA	VALOR EN 203
F1	39
F3	47
F5	55
F7	63

Pruebe el siguiente programa:

```
10 S=PEEK(653):K=PEEK 203:IFK=
  64THEN40
20 K=(1+K)/8-4:PRINT"TECLA"
  K+4*S
30 GOTO 10
40 PRINT"NINGUNA TECLA":GOTO 10
```

Este programa dará los resultados según la siguiente tabla:

Teclas	1	3	5	7
Ninguna tecla	1	2	3	4
SHIFT	5	6	7	8
COMMODORE	9	10	11	12
SHIFT+COMMODORE	13	14	15	16
CTRL	17	18	19	20
SHIFT+CTRL	21	22	23	24
COMMODORE+CTRL	25	26	27	28
SHIFT+COMM+CTRL	29	30	31	32

Nótese que pulsando al mismo tiempo las teclas SHIFT y COMMODORE, cambiará la presentación de mayúsculas a minúsculas y viceversa. Púlselas de nuevo para volver al modo que esté utilizando; la tecla CTRL cuando se pulsa hace que el programa se ejecute a menor velocidad. ■

lectura de "strings"...

(viene de la pág. anterior)

serie de caracteres ASCII prohibidos, tales como:

- retorno de carro
- coma
- dos puntos
- comillas
- etc.

Estas características hacen que la instrucción INPUT # no pueda ser usada cuando se utilizan técnicas de lectura de registro de longitud fija, en los casos en que la longitud del registro sea superior a 79 caracteres o bien el registro contenga caracteres prohibidos (al haberse utilizado técnicas de compactado de campo).

La solución pasa entonces por utilizar la instrucción GET# en un bucle FOR NEXT.

Ejemplo:

```
F$="":REM LG= LONGITUD: FI=
  FICHERO LOGICO
FOR I=1 TO LG
GET# FI,A$ : F$=F$+
  CHR$ (ASC(A$+CHR$ (0))
NEXT I
```

Sin embargo, la instrucción GET# tiene los siguientes inconvenientes:

a) Da cadena nula cuando encuentra un 0 binario.

b) Abre y cierra el IEEE cada vez con la consiguiente pérdida de tiempo.

Para superar estos inconvenientes, se sugiere la siguiente rutina realizada en ensamblador y totalmente relocatable.

SUBRUTINA DE GET DE FICHERO

```
*=$7000 ; dirección de ensamblado
FICH=$ED ; número de fichero
LENST=$EE ; longitud a leer
;
LDA FICH ; carga número de fichero
STA $D2
; mira si longitud a leer es correcta
LDA LENST
BEQ SURT ; si es 0 sale
CMP # 255
BCS SURT ; si es > 254 sale
;
; abertura del IEEE para lectura
;
LDY # 0
LDX $D2 ; número de fichero
JSR $FFC6 ; activación fichero
; (IEEE abierto para el fichero FICH)
;
; bucle de lectura
;
LGET    JSR $FFE4 ; lee un carácter
STA BUFF,Y ; carácter leído a buffer
INY
CPY LENST ; fin de lectura?
BCC LGET
;
; fin lectura
;
; cierre del IEEE
;
SURT    JSR $FFCC ; cierra IEEE
RTS ; retorna
; fin
```

La rutina en código máquina descrita más arriba, trabaja con dos parámetros, que son: número de fichero lógico del fichero a leer y número de caracteres que se desean leer. Estos dos parámetros se colocan antes de llamar a la rutina en las posiciones de página 0 $ED (237 decimal) y $EE (238 decimal). Los caracteres leídos del fichero se colocan en el buffer de trabajo definido al principio, situado entre $7000 y $70FF en hexadecimal (28672 - 28927 decimal).

En próximas apariciones se ahondará sobre soluciones en código máquina a diversos problemas que pueden presentarse en la programación. ■

SUPEREXPANDER

dibujo de tablas gráficas con el superexpander

por N. INCHAURRONDO

Recientemente hemos recibido de Nicolás Inchaurrondo esta colaboración sobre el uso del SUPEREXPANDER. Creemos que el programa es suficientemente explícito por sí mismo para necesitar explicaciones demasiado extensas. Diremos solamente que permite realizar dibujos a partir de una serie de valores que se van entrando (Máx. 40). Sólo una advertencia: este programa únicamente funciona con el superexpander, sin ampliación de memoria de 8 ó 16K. ■

```
READY.

10 PRINT"[ CLR ]"
20 PRINT"[RVSON]        GRAFICAS
30 PRINT" NICOLAS INCHAURRONDO"
40 PRINT"C/ TURO D'EN DENT S/N"
50 PRINT"MALGRAT DE MAR (BARNA)"
60 PRINT"TELEFONO 7610543"
70 PRINT"[COMMT][COMMT][COMMT][COMMT][COMMT]
[COMMT][COMMT][COMMT][COMMT][COMMT][COMMT][COMMT]
[COMMT][COMMT][COMMT][COMMT][COMMT][COMMT][COMMT]
[COMMT][COMMT][COMMT]"
80 PRINT"EL PROGRAMA REPRESEN- "
90 PRINT"TA HASTA 40 VALORES DE"
100 PRINT"CUALQUIER MAGNITUD,    "
110 PRINT"POSITIVOS O NEGATIVOS,"
120 PRINT"ENTEROS O DECIMALES    "
130 PRINT"          [RVSON]TECLEA"
140 GET K$:IFK$=""THEN140
150 DIMD(45):DIMA(45):DIMR(45)
160 K2=1023
170 PRINT"[ CLR ]"
180 PRINT"LOS VALORES NO PUEDEN SER MAS DE 4
0"
190 INPUT"CUANTOS VALORES";P
200 IFP>40ORP<3THENGOTO190
210 PRINT"**************************"
220 FORI=1TOP
230 INPUT"LOS VALORES";A(I)
240 IFA(I)<0THENK2=511
250 IFI=P-1THENPOKE36878,15:POKE36876,225:FO
RT=1TO50 :NEXTT:POKE36878,0
260 NEXTI
270 IFK2=511THENGOSUB650
280 IFK2=511THENGOTO340
290 PRINT"[SHIFE][SHIFE][SHIFE][SHIFE][SHIFE]
[SHIFE][SHIFE][SHIFE][SHIFE][SHIFE][SHIFE][SHIFE]
[SHIFE][SHIFE][SHIFE][SHIFE][SHIFE][SHIFE][SHIFE]
[SHIFE][SHIFE][SHIFE]"
300 PRINT"[RVSON]PREFIERE HISTOGRAMA(H)";
310 PRINT"[RVSON]       O CURVA(C)      "
320 INPUTJ$
330 GOSUB650
340 GRAPHIC2
350 FORI=1TO41
360 DRAW1,0,L2TO1023,L2
370 L2=L2+51
380 IFJ$="H"THENGOTO410
390 DRAW1,L1,0TOL1,1023
400 L1=L1+(1023/(P-1))
410 NEXTI
420 DRAW1,0,K2 TO1023,K2
430 DRAW1,0,K2-10TO1023,K2-10
440 FORQ=1TO40
450 POINT0,H,K2
460 POINT0,0,V
470 H=H+(1023/(P-1))
480 V=V+51
490 NEXTQ
500 FORI=1TOP
510 IFA(I)>=0THENR(I)=K2-A(I)
520 IFA(I)<0THENA(I)=A(I)*-1:R(I)=511+A(I)
530 NEXTI
540 IFJ$="H"THENGOTO940
550 FORI=1TOP
560 POKE36878,15
570 POKE36876,224
580 FORT=1TO15:NEXTT
590 POKE36876,0
600 DRAW1,M,R(I)TOM+(1023/(P-1)),R(I+1)
610 M=M+(1023/(P-1))
620 NEXTI
630 GETA$:IFA$=""THENGOTO630
640 GRAPHIC0:CLR:GOTO10
650 FORI=1TOP
660 D(I)=A(I)
670 IFD(I)<0THEND(I)=D(I)*-1:WH=4
680 NEXTI
690 FORI=1TOP
700 IFD(I)>D(I+1)THEND(I+1)=D(I)
710 NEXTI
720 PRINT"[ CLR ][RVSON] ESCALA DE LA GRAFIC
A "
730 IFWH=0THENJ1=20:GOTO770
740 IFWH=4THENJ1=10:PRINT"VALOR MAXIMO:";D(P
)
750 PRINT"[CRSRD][CRSRD][CRSRD]VALOR ORIGEN:
 0"
760 PRINT"[CRSRD][CRSRD][CRSRD]VALOR MINIMO:
";D(P)*-1:GOTO790
770 PRINT"VALOR MAXIMO:";D(P)
780 PRINT"[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD]VALOR MINIMO: 0"
790 PRINT"[COMM£][COMM£][COMM£][COMM£][COMM£]
[COMM£][COMM£][COMM£][COMM£][COMM£][COMM£][COMM£]
[COMM£][COMM£][COMM£][COMM£][COMM£][COMM£][COMM£]
[COMM£][COMM£][COMM£]":PRINT"UNA DIVISION ="
;D(P)/J1;"U."
800 PRINT"[COMM£][COMM£][COMM£][COMM£][COMM£]
[COMM£][COMM£][COMM£][COMM£][COMM£][COMM£][COMM£]
[COMM£][COMM£][COMM£][COMM£][COMM£][COMM£][COMM£]
[COMM£][COMM£][COMM£]"
810 PRINT"[CRSRD][CRSRD][CRSRD]          [RVSON]
TECLEA"
820 GETA$:IFA$=""THENGOTO820
830 IFD(P)<K2THENGOTO890
840 J5= D(P)/K2
850 FORI=1TOP
860 A(I)=INT(A(I)/J5)
870 NEXTI
880 RETURN
890 J5=K2/D(P)
900 FORI=1TOP
910 A(I)=INT(A(I)*J5)
920 NEXTI
930 RETURN
940 FORI=1TOP
950 POKE36878,15
960 POKE36876,225
970 FORT=1TO100:NEXTT
980 POKE36878,0
990 DRAW1,M,R(I)TOM,K2
1000 M=M+(1023/(P-1))
1010 NEXTI
1020 GETA$:IFA$=""THEN1020
1030 GRAPHIC0:CLR:GOTO10
READY.
```

COLABORACIONES

varios programas útiles

gráficos en alta resolución con un programa en BASIC

En la figura 1 se lista un programa que sirve como ejemplo de dibujo en alta resolución sin el uso de ningún cartucho especial.

definición de teclas de función

J. L. Martínez, de Málaga, nos manda el programa de la figura 2 que permite definir una secuencia de caracteres para cada tecla de función de F1 a F8. Después de cargar y rodar el programa, éste preguntará F1? Como respuesta debe teclearse la sucesión de signos que se deban asignar a esta tecla, pulsando RETURN para terminar. A continuación se repetirá la misma operación con todas las teclas de función hasta F8. Al terminar el programa desaparece pudiéndose cargar otro programa.

```
READY.

5 F=0:C=PEEK(55)-120:IFC<0THENC=C+256:F=-1
10 D=PEEK(56)+F:POKE55,C:POKE56,D:CLR
15 S=828:I=146:GOSUB100
20 DATA32,198,3,165,55,133,251,133,253,165,5
6,133,252,133,254,169,49,133,0,169
25 DATA133,133,1,169,13,32,210,255,169,70,32
,210,255,165,0,32,210,255,169,61
30 DATA32,210,255,169,63,32,210,255,169,32,3
2,210,255,32,207,255,72,160,0,165
35 DATA1,145,55,104,32,198,3,201,13,240,14,2
01,95,208,2,169,13,145,55,32
40 DATA207,255,76,124,3,230,0,165,0,41,1,208
,10,24,165,1,105,4,133,1
45 DATA76,170,3,56,165,1,233,3,133,1,165,0,2
01,57,144,163,120,169,L0,141
50 DATA20,3,169,H0,141,21,3,88,169,0,133,0,3
2,68,198,76,116,196,166,55
55 DATA208,2,198,56,198,55,96
60 S=PEEK(55)+256*PEEK(56):I=119:GOSUB100
65 SYS(828)
70 DATA165,0,240,59,160,0,177,251,32,L99,H0,
176,12,165,55,197,251,208,21,165
75 DATA56,197,252,208,15,169,0,133,0,165,253
,133,251,165,254,133,252,76,191,234
80 DATA166,198,177,251,157,119,2,230,198,32,
L111,H0,165,198,201,11,144,204,230,0
85 DATA76,191,234,165,215,32,L99,H0,176,3,76
,191,234,165,8,41,1,208,247,160
90 DATA0,177,251,197,215,208,6,32,L111,H0,76
,L6,H0,32,L111,H0,76,L81,H0,201
95 DATA133,144,6,201,141,176,2,56,96,24,96,1
66,251,208,2,198,252,198,251,96
100 F=0:FORD=STOS+I:READA$:IFASC(A$)<58THENA
=VAL(A$):GOTO115
105 IFASC(A$)=76THENA=VAL(RIGHT$(A$,LEN(A$)-
1))+PEEK(55):IFA>255THENA=A-256:F=1
110 IFASC(A$)=72THENA=VAL(RIGHT$(A$,LEN(A$)-
1))+PEEK(56)+F:F=0
115 POKED,A:NEXT:RETURN
READY.
```

Fig. 2

```
READY.

1000 REM INICIALIZACION DE ALTA RESOLUCION
1005 PRINT"[ BLK ]":REM CTRL Y BLK
1010 PRINT"[ CLR ]":POKE36879,79
1020 IFPEEK(36869)=253THEN1080
1030 POKE36869,253:POKE36867,PEEK(36867)OR12
8
1040 POKE55,0:POKE56,19:POKE51,0:POKE52,19
1050 CLR:S=32768:T=5120
1060 PRINT"[ CLR ][CRSRD][CRSRD][CRSRD] ESTO
Y INICIALIZANDO"
1070 FORI=0TO255*8+7:POKEI+T,PEEK(I+S):NEXT
1080 REM FIN DE INICIALIZACION
2000 REM SUBRUTINA USUARIO
2010 PRINT"[ CLR ]"
2020 PRINT"FIGURAS";TAB(15);"COCOS"
2022 PRINT"PRECIO";TAB(15);"PRECIO":PRINT:PR
INT
2025 READLB:LT=LB+100:READRB:RT=RB+100
2030 FORI=0TO10:X=LT-10*I:Y=RT-10*I
2040 X$=RIGHT$("  "+STR$(X),3)
2042 Y$=RIGHT$("  "+STR$(Y),3)
2050 PRINTX$;TAB(17);Y$:NEXT
2058 REM HACER EJE VERTICAL
2060 FORI=5TO15:POKE7680+22*I+3,115:NEXT
2070 POKE7680+15*22+3,91:REM PONER LA UNION
DE LOS EJES
2078 REM HACER EJE HORIZONTAL
2080 FORI=4TO16:POKE7680+15*22+I,114:NEXT
2088 REM HACER EJE VERTICAL
2090 FORI=5TO15:POKE7680+22*I+16,115:NEXT
2100 PRINT"[CRSRD]";TAB(3);"1234567890123"
3000 REM PRIMER GRAFICO
3010 FORX=28TO132
3020 READ D:IFD=0THEN3060
3030 Y=INT((LT+55-D)*8/10)
3040 GOSUB8000
3050 NEXT
3060 REM PRIMER PLOT
4000 REM SEGUNDO GRAFICO
4010 FORX=28TO132
4020 READ D:IFD=0THEN4060
4030 Y=INT((RT+55-D)*8/10)
4040 GOSUB8000
4050 NEXT
4060 REM SEGUNDO GRAFICO
5000 GOTO9000
6000 DATA60,100
6010 REM PRIMER GRAFICO
6020 DATA100,102,105,108,110,115,112,118
6022 DATA120,122,118,120,120,124,126,130
6024 DATA128,126,128,126,125,124,126,122
6026 DATA 122,126,128,130,132,136,140,142
6028 DATA144,142,139,138,136,137,135,138
6030 DATA136,134,132,132,130,128,130,134
6032 DATA 0
6100 REM SEGUNDO GRAFICO
6110 DATA 102,104,110,106,108,110,110,108
6112 DATA 110,112,113,115,118,114,116,118
6114 DATA 120,122,124,120,118,116,114,110
6116 DATA 110,112,116,118,120,122,124,120
6118 DATA 120,118,118,116,114,113,116,115
6120 DATA 114,110,112,108,106,110,114,116
6122 DATA 0
8000 REM SUBRUTIPLOT
8010 X%=X/8:Y%=Y/8:P=X%+Y%*22+7680
8020 Q=PEEK(P):IFQ>=128THEN8050
8030 CN=CN+1:S=5120+(127+CN)*8:T=5120+Q*8
8040 FORI=0TO7:POKES+I,PEEK(T+I):NEXT
8045 Q=127+CN:POKEP,Q
8050 C=5120+Q*8+(YAND7)
8055 POKEC,PEEK(C)OR(2↑(7-(XAND7)))
8060 RETURN
9000 REM FIN DE RUTINA
9010 PRINT"[HOME ][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD][CRSRD][CRSRD][CRSRD][CRSRD][CRSRD]
[CRSRD][CRSRD]";
9015 PRINT"PULSAR UNA TECLA PARA ACABAR"
9020 GET A$:IF A$=""THEN9020
9030 END
READY.
```

Fig. 1

un programa para la impresora

Juan R. Torán Torrescassana, de Madrid, nos ha enviado un interesante y decorativo programa para utilizar la impresora del VIC-20. Este programa para la impresora que se lista en la figura 3 permite la realización de «posters» como el de la figura 4. El texto del fondo se puede variar.

```
READY.

10 PRINT"[ CLR ]"
60 INPUT"PALABRA:";A$:L=LEN(A$)
65 OPEN1,4:CMD1
70 DIMT$(120):FORI=1TO10:PRINT:NEXTI
100 FORJ=0TOINT(60/L)
110 FORI=1TOL
120 T$(J*L+I)=MID$(A$,I,1)
130 NEXTI:NEXTJ
140 C=0
200 A1=1:P=1:C=C+1:IFC=37THEN999
205 PRINT
210 READA:A1=A1+A:IFP=1THEN300
240 FORI=1TOA:PRINT" ";:NEXTI:P=1:GOTO400
300 FORI=A1-ATOA1-1:PRINTT$(I);:NEXTI:P=0
400 IFA1>60THEN200
410 GOTO210
600 DATA60,1,12,26,9,12,3,3,24,17,8,4,6,23,2
1,6,4,6,22,12,5,6,5
610 DATA4,6,21,11,8,6,4,4,6,21,10,10,5,4,4,6
,21,9,11,5,4
620 DATA4,6,21,8,11,6,4,4,6,21,7,11,7,4,4,6,
21,6,11,8,4
630 DATA4,6,19,1,1,5,11,9,4,4,6,19,1,1,5,10,
10,4,4,6,18,2,1,6,8,11,4
640 DATA4,6,17,3,1,7,5,13,4,4,6,15,5,2,23,5,
1,29,5,17,8
650 DATA1,29,9,9,12,1,13,5,40,1,1,13,5,40,1,
4,6,13,3,10,6,12,5,1
660 DATA5,6,11,3,11,6,14,3,1,5,6,11,3,11,6,1
5,2,1
670 DATA6,6,9,3,12,6,16,1,1,6,6,9,3,12,6,7,1
,10
680 DATA7,6,7,3,13,6,6,2,10,7,6,7,3,13,14,10
,8,6,5,3,14,6,6,2,10
690 DATA8,6,5,3,14,6,7,1,10,9,6,3,3,15,6,16,
1,1
700 DATA9,6,3,3,15,6,15,2,1,10,6,1,3,16,6,14
,3,1,10,10,16,6,12,5,1
710 DATA11,8,13,27,1,11,8,13,27,1,60
999 FORI=1TO10:PRINT:NEXTI
1000 PRINT#1:CLOSE1
READY.
```

Fig. 3

```
CLUB COMMODORECLUB COMMODORECLUB COMMODORECLUB COMMODORECLUB
C            ECLUB COMMODORECLUB COMMOD         OMMODORECLUB
CLU        ORECLUB COMMODORECLUB CO                 DORECLUB
CLUB      DORECLUB COMMODORECLUB                      RECLUB
CLUB      DORECLUB COMMODORECLUB            UB CO      ECLUB
CLUB      DORECLUB COMMODORECLU           CLUB COM      CLUB
CLUB      DORECLUB COMMODORECLU          ECLUB COMM     CLUB
CLUB      DORECLUB COMMODORECLU         RECLUB COMM     CLUB
CLUB      DORECLUB COMMODORECLU        ORECLUB COM      CLUB
CLUB      DORECLUB COMMODORECLU       DORECLUB CO       CLUB
CLUB      DORECLUB COMMODORECLU      ODORECLUB C        CLUB
CLUB      DORECLUB COMMODOREC U     MODORECLUB          CLUB
CLUB      DORECLUB COMMODOREC U     MODORECLUB          CLUB
CLUB      DORECLUB COMMODORE  U      ODORECLU           CLUB
CLUB      DORECLUB COMMODOR   U       DOREC             CLUB
CLUB      DORECLUB COMMOD     UB                       ECLUB
C                             UB CO                 DORECLUB
C                             UB COMMOD         OMMODORECLUB
C             CLUB                                         B
C             CLUB                                         B
CLUB      DORECLUB COMM   RECLUB COM      CLUB COMMODO     B
CLUB       ORECLUB COM   ORECLUB COM      CLUB COMMODORE   B
CLUB       ORECLUB COM   ORECLUB COM      CLUB COMMODOREC  B
CLUB C      RECLUB CO   DORECLUB COM      CLUB COMMODORECL B
CLUB C      RECLUB CO   DORECLUB COM      CLUB CO MODORECLUB
CLUB CO      ECLUB C   ODORECLUB COM      CLUB C  MODORECLUB
CLUB CO      ECLUB C   ODORECLUB COM              MODORECLUB
CLUB COM      CLUB    MODORECLUB COM      CLUB C  MODORECLUB
CLUB COM      CLUB    MODORECLUB COM      CLUB CO MODORECLUB
CLUB COMM      LUB   MMODORECLUB COM      CLUB COMMODORECL B
CLUB COMM      LUB   MMODORECLUB COM      CLUB COMMODOREC  B
CLUB COMMO      U   OMMODORECLUB COM      CLUB COMMODORE   B
CLUB COMMO          OMMODORECLUB COM      CLUB COMMODO     B
CLUB COMMOD        COMMODORECLUB                           B
CLUB COMMOD        COMMODORECLUB                           B
CLUB COMMODORECLUB COMMODORECLUB COMMODORECLUB COMMODORECLUB
```

Fig. 4

un calendario por pantalla

También nos ha hecho llegar J. L. Martínez, un programa para la elaboración de un calendario el cual se lista en la figura 5. El programa es autoexplicativo.

```
10 REM CALENDARIO PARA EL VIC
20 REM POR JOSE LUIS MARTINEZ
30 YY=1900
40 CC=1
50 POKE36879,8
60 B1$="CALENDARIO ":B2$="  PERPETUO     ":B
3$=" PARA EL VIC "
70 PRINT"[ CLR ]":PRINT:PRINT"[RVSON][ GRN ]
   [RVSOF]"
80 PRINT"[RVSON]       [RVSOF]"
90 PRINT"[RVSON]        [RVSOF]"
100 FORI=1TO13:PRINTTAB(I-1)"[RVSON]"MID$(B3
$,I,1);" ";MID$(B2$,I,1);" ";MID$(B1$,I,1);"
[RVSOF]"
110 NEXT
120 PRINT"[ WHT ]POR:[ GRN ]";SPC(I-4);"[RVSOF]
[RVSON]    [RVSOF]"
130 PRINTSPC(I);"[RVSOF][RVSON]    [RVSOF]"
140 PRINT"[ WHT ] JOSE LUIS[ GRN ]";SPC(I-11
);"[RVSOF][RVSON]    "
150 PRINT"[HOME ]"
160 GETA$:IFA$=""THEN160
170 IFA$="N"THEN310
180 PRINT"[ GRN ][ CLR ]ESTE PROGRAMA HACE U
N CALENDARIO POR UN"
190 PRINT"A[RVSON]N[RVSOF]O[ RED ][RVSON]DES
DE";YY;"[RVSOF]."
200 PRINT:PRINT"
      ";
210 PRINT:PRINT"[ BLU ]PARA ENTRAR OTRO CALE
NDARIO PULSAR[RVSON]F1[RVSOF]."
220 PRINT:PRINT"[ YEL ]EL A[RVSON]N[RVSOF]O
EMPIEZA SIEMPRE POR ENERO. PARA"
230 PRINT"VER LOS MESES SIGUIENTES PULSE OTR
A TECLA.";
240 PRINT"[ PUR ](EN DICIEMBRE EL CALENDARIO
 CONTINUA PARA EL SIGUIENTE A[RVSON]N[RVSOF]
O)":PRINT
250 INPUT"[ WHT ]A[RVSON]N[RVSOF]O BUSCADO";
Y
260 DA=365:IFY/4=INT(Y/4)THEN DA=366
270 H=0:DB=DA
280 IFY>YYTHEN340
290 RESTORE
300 K=0
310 PRINT"[ CLR ][ WHT ]NO PUEDO COMENZAR AN
TES";YY
320 POKE36879,8
330 GOTO 250
340 Z=Y-YY:C=CC
350 FORI=1TOZ
360 C=C+1:YX=YY+I
370 IFYX/4=INT(YX/4)THENC=C+1
380 IFC>=7THENC=C-7
390 NEXTI
```

Fig. 5

SOFTWARE DE BASE (II)

método de selección



por E. MARTÍNEZ DE CARVAJAL

Continuando con el estudio de los métodos de ordenación iniciado en nuestro número anterior, hoy le toca el turno al método de selección.

La idea básica de este método es —como indica su nombre— ir seleccionando los elementos de clave más pequeña que se van colocando en una lista de salida que va creciendo de forma ordenada.

Al principio de cada pasada se supone que el primer elemento es el de clave más pequeña y se va comparando con todos los demás, buscando alguno que sea menor. Si se encuentra alguno, se toma éste como el menor de la lista, el cual se transfiere a la lista de salida, poniéndose en su lugar en la lista original un valor ficticio que sea superior a todos los demás (este valor habrá que adaptarlo al tipo de los elementos que se utilicen). El procedimiento finaliza cuando se han seleccionado todos los elementos de la lista, o lo que es lo mismo, cuando hayamos hecho tantas pasadas como número de elementos y como en cada pasada se ha de comparar un elemento con todos los demás, nos encontramos nuevamente con un método de ordenación insensible al estado original de la tabla y con tiempos de ordenación iguales a N*(N-1), siendo N el número de elementos de la tabla, lo cual nos da un rendimiento muy similar al método de contaje. En el listado 1 se presenta una rutina de ordenación basada en este método.

CLASIFICACIÓN POR INTERCAMBIO

La familia de los métodos de clasificación que se denominan «por intercambio», está basada en el intercambio sistemático de posiciones cada vez que se encuentra un par de elementos desordenados entre sí, hasta que no exista ninguno de estos pares.

El proceso se efectúa del siguiente modo: en la primera pasada el primer miembro de la lista se compara con su inmediato sucesor; si éste es más pequeño, se intercambia. El elemento mayor, que ahora está en la segunda posición, se compara con el que está en la tercera y, si aquél es mayor, se intercambian, y así sucesivamente. Cuando se compara la posición N-1 con la N, finaliza la pasada, repitiéndose el proceso. El efecto de esto es que los elementos de clave más alta se van moviendo hacia abajo.

El método más representativo de esta familia es el conocido con el nombre de la «burbuja» («bubble» en inglés), cuyo nombre se deriva precisamente de que los elementos más grandes se «hunden» en la lista, mientras que los más pequeños «flotan». Este método es muy conocido a pesar de que sus resultados no son muy buenos. Sin embargo, es un método inteligente y que se puede mejorar mucho con facilidad.

No es difícil ver que después de cada pasada, todos los elementos que están por encima del último intercambio —incluyendo este último— no necesitan examinarse en las pasadas sucesivas y además, si en alguna pasada no se realiza ningún intercambio, quiere decir que la tabla ya está ordenada, con lo que el tiempo de ordenación dependerá del estado inicial de la tabla. Otra ventaja es que no se necesita otra tabla de salida, ya que la ordenación se realiza sobre la original.

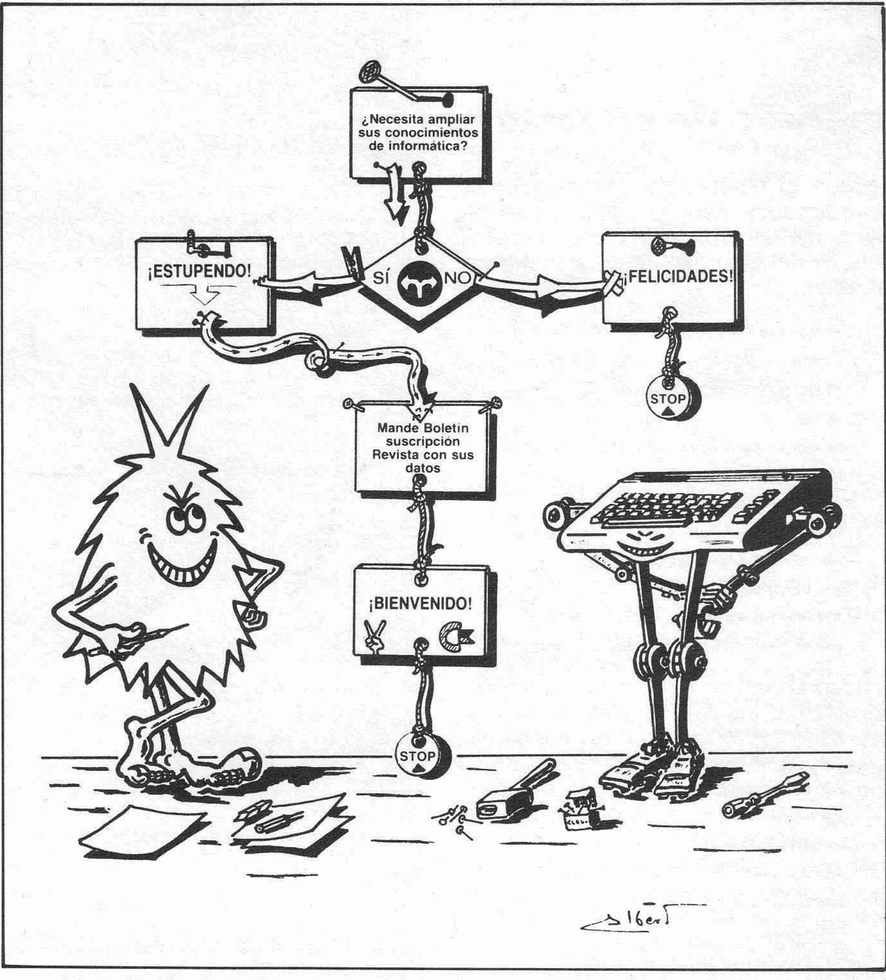
Finalmente, si en vez de realizar las pasadas siempre en un mismo sentido, se van alternando, se consigue

(termina en la pág. 8)

```
10 REM *** SELECCION ***
20 REM ---------------------------------
30 REM METODO DE ORDENACION
40 REM
50 REM ERNESTO MTNZ. DE CARVAJAL HEDRICH
60 REM
70 REM 13-MAR-83
80 REM ---------------------------------
100 INPUT "NUMERO ELEMENTOS";M9
110 DIM W(M9),O(M9),A1(M9)
120 FOR I=1 TO M9
130 W(I)=INT(RND(0)*900+100):O(I)=W(I)
140 NEXT I
150 FOR I=1TOM9:PRINTW(I):NEXTI
160 TI$="000000"
200 REM *** RUTINA DE ORDENACION ***
220 TI$="000000"
280 FORI=1TOM9
290 MIN=I
300 FORJ=1TOM9
310 IFW(J)<W(MIN)THENMIN=J
320 NEXTJ
330 K=K+1:A1(K)=W(MIN):W(MIN)=99999999
340 NEXTI
440 REM FIN ORDENACION
3500 T$=TI$
3510 OPEN 1,4
3520 PRINT#1,"ORDENACION DE UNA TABLA DE "M9
" ELEMENTOS POR EL METODO DE SELECCION"
3530 PRINT#1,"TIEMPO DE ORDENACION : "LEFT$(
T$,2)":"MID$(T$,3,2)":"RIGHT$(T$,2)
3540 PRINT#1
3550 PRINT#1,"TABLA INICIAL  TABLA FINAL"
3560 PRINT#1,"[SHIF*][SHIF*][SHIF*][SHIF*][SHIF*]
[SHIF*][SHIF*][SHIF*][SHIF*][SHIF*][SHIF*][SHIF*]
[SHIF*]  [SHIF*][SHIF*][SHIF*][SHIF*][SHIF*]
[SHIF*][SHIF*][SHIF*][SHIF*][SHIF*][SHIF*]"
3570 FOR I=1TOM9
3580 PRINT#1,"     "O(I)"          "A1(I)
3590 NEXT I
4000 PRINT#1
4010 CLOSE1
4020 STOP
READY.
```

Listado 1

clave para interpretar los listados de CLUB COMMODORE



Todos los listados que se publican en esta Revista han sido ejecutados en el modelo correspondiente de la gama de ordenadores COMMODORE. Para facilitar la edición de los mismos en la Revista y para mejorar su legibilidad por parte del usuario, se les ha sometido a ciertas modificaciones mediante un programa escrito especialmente para ello. Para los programas destinados a los ordenadores VIC-20 y COMMODORE 64, en los que se usan frecuentemente las posibilidades gráficas del teclado, se han sustituido los símbolos gráficos que aparecen normalmente en los listados por una serie de letras entre corchetes [ ] que indican la secuencia de teclas que se deben pulsar para obtener el carácter deseado. A continuación se da una tabla para aclarar la interpretación de las indicaciones entre corchetes:

[CRSRD] = Tecla cursor hacia abajo (sin SHIFT)
[CRSRU] = Tecla cursor hacia arriba (con SHIFT)
[CRSRR] = Tecla cursor a la derecha (sin SHIFT)
[CRSRL] = Tecla cursor a la izquierda (con SHIFT)
[HOME] = Tecla CLR/HOME (sin SHIFT)
[CLR] = Tecla CLR/HOME (con SHIFT)

Las indicaciones [BLK] a [YEL] corresponden a la pulsación de las teclas de 1 a 8 junto a la tecla CTRL. Lo mismo sucede con [RVSON] y [RVSOF] respecto a la tecla CTRL y las teclas 9 y 0.

El resto de las indicaciones constan de la parte COMM o SHIF seguidas de una letra, número o símbolo — por ejemplo [COMM+] o [SHIFA] —. Esto indica que para obtener el gráfico necesario en el programa deben pulsarse simultáneamente las teclas COMMODORE (la que lleva el logotipo) o una de SHIFT y la tecla indicada por la letra, el número o el símbolo, en el ejemplo anterior: COMMODORE y + o SHIFT y A, respectivamente. ■

método de selección

(viene de la pág. 6)

reducir el número de comparaciones y de pasadas. Esta modificación da lugar al método conocido como «la cocktelera» del cual damos un ejemplo en el listado 2.

```
10 REM *** COCKTELERA ***
20 REM ---------------------------------
30 REM METODO DE ORDENACION
40 REM
50 REM ERNESTO MTNZ. DE CARVAJAL HEDRICH
60 REM
70 REM 26-FEB-83
80 REM ---------------------------------
100 INPUT "NUMERO ELEMENTOS";M9
110 DIM W(M9),O(M9)
120 FOR I=1 TO M9
130 W(I)=INT(RND(0)*900+100):O(I)=W(I)
140 NEXT I
150 FOR I=1TOM9:PRINTW(I):NEXTI
160 TI$="000000"
200 REM *** RUTINA DE ORDENACION ***
220 TI$="000000"
280 J=M9-1:J2=1
300 S1=0
310 FOR I=J2 TO J
320 IF W(I)<=W(I+1) THEN 350
330 A1=W(I)
340 W(I)=W(I+1):W(I+1)=A1:S1=1:J3=I
350 NEXTI
355 J=J3
360 IF S1=0 THEN 440
370 S1=0
380 FOR I=J TO J2+1 STEP-1
390 IFW(I)>=W(I-1) THEN 420
400 A1=W(I)
410 W(I)=W(I-1):W(I-1)=A1:S1=1:J3=I
420 NEXT I
425 J2=J3
430 IF S1=1 THEN300
440 REM FIN ORDENACION
3500 T$=TI$
3510 OPEN1,4
3520 PRINT#1,"ORDENACION DE  UNA TABLA DE "M
9" ELEMENTOS POR EL METODO DE LA COCKTELERA"
3530 PRINT#1,"TIEMPO DE ORDENACION : "LEFT$(
T$,2)":"MID$(T$,3,2)":"RIGHT$(T$,2)
3540 PRINT#1
3550 PRINT#1,"TABLA INICIAL  TABLA FINAL"
3560 PRINT#1,"[SHIFR][SHIFR][SHIFR][SHIFR][SHIFR]
[SHIFR][SHIFR][SHIFR][SHIFR][SHIFR][SHIFR][SHIFR]
[SHIFR]  [SHIFR][SHIFR][SHIFR][SHIFR][SHIFR]
[SHIFR][SHIFR][SHIFR][SHIFR][SHIFR][SHIFR]"
3570 FOR I=1TOM9
3580 PRINT#1,"     "O(I)"        "W(I)
3590 NEXT I
4000 PRINT#1
4010 CLOSE1
4020 STOP
READY.
```

Listado 2

A continuación se dan los tiempos de ordenación para los dos métodos y para tablas de 50 y 100 elementos generadas aleatoriamente

Método	Tiempo para 50 elem.	Tiempo para 100 elem.
Selección	00:00:19	00:01:14
Cocktelera	00:00:19	00:01:10

problemas con la expansión de memoria a más de 8 K

por P. MASATS

Aunque este tema ya se ha tocado con más extensión en otros números (para información de «revistófilos» y lectores veteranos, en el número 1 página 6), la gran cantidad de consultas recibidas nos hace creer que debemos explicar el problema de la expansión de memoria a más de 8K.

Cuando se amplía la memoria del VIC a más de ocho Kbytes (es decir, cuando se inserta un cartucho de 8 ó 16K) todos los programas que efectúan POKES en pantalla dejan de funcionar por razones misteriosas. La razón de ello es que, por necesidades internas del sistema operativo, las matrices de memoria de pantalla y de color se sitúan en otro sitio, y los POKES que servían anteriormente, se efectúan en sitios que no tienen nada que ver con el procesador de video.

Si en los programas que escribamos en el futuro y que puedan necesitarla, incorporamos la siguiente rutina y tenemos la precaución de efectuar los pokes no en la forma POKE7680,X sino POKES,X tendremos solucionado el problema. La rutina es:

```
100 S=7680:C=38400:IF PEEK(644)>32 THEN S=4096:C=37888
```

Como más de uno debe haber adivinado, la variable C es la posición de la memoria de color, y las posiciones de la matriz de pantalla y la de color con más de 8K son 4096 y 37888 respectivamente.

NOTAS

una mejora significativa del programa para hacer listados

La adición de las siguientes líneas incrementan considerablemente la velocidad de ejecución del programa para confeccionar listados que publicamos en el número 6:

```
120 GOSUB 165
165 IFASC(A$)> # ¡ANDASC
   (A$)<144THENRETURN
167 IFASC(A$)>223THENRETURN
```



PROGRAMAS EN DISCO

D-1001 Agenda 5.000
Permite contener 4 páginas de información de 114 personas, entidades, etc. Pudiendo añadir, cambiar o borrar la información que se desee.
En castellano, necesita ampliación de 8K. Presentado en caja.

D-1002 QSL 3.000
Registro de contactos para radioaficionados, versión en disco.
Necesita ampliación de 8K. Presentado en caja.

D-1003 Test Demo 3.000
Programa de test para la unidad de disco. Nota: este disco es el que se suministra con la unidad de disco VIC-1540.

D-1004 Assembler 5.000
Editor y compilador en BASIC para rutinas en lenguaje máquina del 6502. Necesita ampliación de 3K. Acompañado de manual en inglés.

D-1005 English language 2.500
Versión en disco del programa en cinta, dado que al manejar ficheros resulta mucho más rápida su utilización.
Necesita ampliación de 8K. Presentado en caja.

D-1006 Quiz Master 2.500
Versión en disco del programa en cinta. Necesita ampliación de 8K. Presentado en caja.

D-1007 Matemáticas 1 (nivel BUP) 2.500
Versión en disco del programa en cinta. Necesita ampliación de 8K. Presentado en caja.

3301 Simplicalc 13.000
Hoja de Trabajo Electrónica 60 columnas por 100 filas Max. Necesita ampliación de 16K.

3304 Vic File 13.000
Bases de datos para el VIC-20 con posibilidad de definir el formato de los campos. Necesita ampliación de 16K. Ideal para fichero de clientes.

3305 Vic Writer 13.000
Proceso de texto muy potente y rápido. Necesita ampliación de 8K ó 16 K.

PROGRAMAS EN CINTA

Programas Técnicos

C-128 Programación lineal 1.000
Método simplex. Cálculo del valor de las variables que satisfaciendo las restricciones hacen máxima o mínima una función. En castellano. Presentado en caja.

C-129 Matrices 1.000
Suma, resta, multiplicación, multiplicación por un escalar e inversión de matrices. En castellano. Presentado en caja.

C-131 Regresiones I 1.000
Contenido: Regresión lineal: Cálculo por mínimos cuadrados de la recta que se ajusta mejor a una nube de puntos.
Regresión múltiple: Variable dependiente en función de N variables independientes de grado 1. En castellano. Presentado en caja.

C-132 Regresiones II 1.000
Contenido: Regresión de orden N: Variable dependiente en función de variable independiente de grado N.
Regresión exponencial: Ajuste nube de puntos a una curva exponencial.
Regresión geométrica: Ajuste de una nube de puntos a una curva geométrica. En castellano. Presentado en caja.

C-133 Estadística I 1.000
Distribución normal, Poisson, binomial, chi-cuadrado, Student, F de Snedecor. En castellano. Presentado en caja.

C-134 Estadística II 1.000
Contenido: Cálculo de la media, varianza y desviación tipo, tanto de la muestra como de la población, estando los datos agrupados o no. Test de chi-cuadrado y test de Student. En castellano. Presentado en caja.

C-135 Sistemas 1.000
Contenido: Resolución de sistemas de N ecuaciones con N incógnitas.
Resolución de ecuaciones de grado 2 dando las soluciones tanto reales como complejas.
Cálculo de permutaciones y de combinaciones. En castellano. Presentado en caja.

C-137 Integración 1.000
Cálculo de integrales por el método de Gauss.
Cálculo de derivadas de una función en un punto. Interpolaciones tanto lineales como curvilíneas. En castellano. Presentado en caja.

Programas Educativos

C-125 Hangmath 1.500
Se trata de averiguar la multiplicación entre 2 números (tanto los números como la multiplicación y el resultado final), procurando no ser ahorcado. Las instrucciones son en castellano.

C-140 Skymath 1.500
Sumas y restas para chicos muy jóvenes saliendo naves espaciales para motivarlos. Una manera de aprender jugando. 3K.

C-141 Space Division 1.500
20 preguntas sobre divisiones muy sencillas. En función de los resultados correctos, despega un cohete que llegará más o menos lejos. 3K.

C-143 English Language 2.000
Especialmente adecuado para las personas que, poseedoras de un buen nivel de inglés, quieren alcanzar cotas superiores. Necesita ampliación de 8K. Presentado en caja y acompañado de su correspondiente manual.

C-145 Mastermind 3.000
Participe en el famoso concurso de la BBC, de preguntas y respuestas sobre temas variados.
Este cassette va acompañado del sistema operativo y de un file de demostración. Necesita ampliación de 8K. Presentado en caja.

C-146 Matemáticas I (nivel BUP) 2.000
Contiene sistema operativo y 7 cuestionarios sobre diferentes temas abarcando geometría, aritmética, matemáticas generales, álgebra, etc. Necesita ampliación de 8K ó 16K. Presentado en caja.

PROGRAMAS COMERCIALIZADOS POR "MICROELECTRÓNICA Y CONTROL, S.A."
Puede encontrarlos en su distribuidor más próximo

microelectrónica y control s a







Este es el nuevo ordenador personal COMMODORE 64. Un gigante de 40 cm, con un precio casi tan pequeño como su tamaño.

Nadie hasta ahora había logrado ofrecerle 64 K de memoria, 40 columnas en pantalla, 8 sprites y un sonido de auténtica maravilla por sólo 110.000,— ptas. Claro que tampoco todo el mundo es el líder mundial en microordenadores.

COMMODORE sabe perfectamente que para seguir siendo el número uno, tiene que estar constantemente en vanguardia.
De calidad. De precios. De todo. Para ello investigamos constantemente.

Afortunadamente nuestra labor se ve plenamente recompensada cuando vemos, como lo demuestra el cuadro comparativo, que nuestro más directo competidor cuesta nada menos que un 100% más caro. Y ello sin reunir todos los adelantos técnicos del COMMODORE 64.

---

1. Capacidad total de memoria RAM de 64 K. Interpretador BASIC extendido y sistema operativo residentes en ROM.

2. Dotado del más potente chip sintetizador de sonido diseñado hasta hoy, el COMMODORE 64 ofrece 3 voces totalmente independientes con una gama de 9 octavas. El programa puede controlar la envolvente, la afinación y la forma de onda de cada voz,

convirtiendo al COMMODORE 64 en el mejor simulador de instrumentos.

3. Conectable directamente a toda una gama de periféricos, incluyendo unidad de discos, impresora de matriz de puntos o de margarita, plotter, comunicaciones locales y remotas..., y mucho más.

4. Pantalla de alta resolución en color con 320 × 200 puntos directamente direccionables. Capacidad en modo carácter de 25 líneas por 40 columnas.

5. El chip de video, único en su género, permite el uso de 8 «Sprites» (figuras móviles en alta resolución y color). Los «Sprites» pueden moverse independientemente por programa de «pixel» en «pixel».

6. A cada «Sprite» se le asigna por programa un nivel de prioridad en caso de cruce con otro, consiguiendo efectos tridimensionales, existiendo también detección automática de colisiones.

7. Teclado profesional con mayúsculas y minúsculas, más 62 caracteres gráficos, todos ellos disponibles en el teclado y visualizables en 16 colores, en forma normal o bien en video invertido.

8. Encontrará a su disposición una completa gama de programas profesionales, incluyendo proceso de textos, sistemas de información, modelos financieros, contabilidad y muchas más aplicaciones.

9. Están en fase de desarrollo asimismo otros lenguajes tales como LOGO, UCSD PASCAL, COMAL, ASSEMBLER, etc. Todos los programas existentes de la gama COMMODORE, desde el VIC-20 hasta los modelos CBM pueden ser adaptados fácilmente.

10. Posibilidad de inserción de cartuchos con programas grabados en ROM, tanto profesionales como para educación y ocio.

11. Opción de un segundo procesador Z-80 para trabajar con sistema operativo CP/M (R).

EL COMMODORE 64 Y SU MAS DIRECTO COMPETIDOR

OPCION DE BASE	COMMODORE 64	Más directo competidor
Precio	110.000,— ptas.	El doble
Memoria usuario	64 K	48 K
Teclado profesional	SI	SI
Teclado con caracteres gráficos	SI	NO
Minúsculas	SI	NO
Teclas de función	SI	NO
Máxima capacidad disco	170 K a 1 M	143 K
AUDIO		
Generador de sonido	SI	SI
Sintetizador de música	SI	NO
Salida HI-FI	SI	NO
VIDEO		
Salida monitor	SI	SI
Salida para TV	SI	EXTRA
PERIFERICOS		
Cassette	SI	SI
Periféricos inteligentes	SI	SI
Bus serie	SI	NO
SOFTWARE		
Opción CP/M (R)	SI	SI
Ranura cartucho externo	SI	NO

commodore
COMPUTER

PARA MAS INFORMACION
DEL COMMODORE 64,
LLAMAR O ESCRIBIR A:
MICROELECTRONICA Y CONTROL
c/ Taquígrafo Serra, 7, 5.º. Barcelona-29
Tel. (93) 250 51 03
c/ Princesa, 47, 3.º, G. Madrid-8
Tel. (91) 248 95 70

Nombre ....
Dirección ....
Tel. ....
Población ....



Programas Aplicaciones

C-130 Caja 2.000
Dentro de una cuenta bancaria permite abrir hasta 100 subcuentas con su correspondiente debe y haber. Preparado para funcionar si se desea con impresora, unidad de discos y evidentemente cassette. Necesita ampliación 16K. Presentado en caja.

C-136 Dieta 2.000
En función de la edad, sexo, altura, configuración y actividad desempeñada por la persona, da el peso ideal. Pregunta si se desea seguir un régimen, y en función de la actividad física desarrollada da el peso que se debe tener y la pérdida diaria que se debe alcanzar hasta llegar a dicho peso óptimo. Está en castellano y necesita ampliación de 8K. Presentado en caja.

C-139 Vicalc 1.500
El VIC-20 se comporta como una calculadora y utiliza la notación polaca inversa. Manual de instrucciones en castellano.

C-142A Interface de RTTY y CW y cassette de CW . 25.000
Especial para radioaficionados.

C-142B Cassette de RTTY 2.500
Para utilizarlo es necesario el interface de RTTY y CW. Especial radioaficionados.

C-144 Quiz Master 2.000
Programa para preparación de cuestionarios. Especialmente adecuado para escuelas. En castellano. Necesita ampliación de 8K.

PROGRAMAS EN CARTUCHO

1901 Avenger 4.500
Típica batalla de marcianitos. Sus enemigos avanzan en fila mientras disparan teniendo como protecciones su rapidez y velocidad de disparo y unos cuantos bunkers.

1902 Star Battle 4.500
Batalla de marcianitos con ataque de ovnis estilo «stukas». Se juega con teclado o con Joystick.

1904 Super Slot 4.500
Juego en cartucho que le permitirá jugar con una máquina tragaperras cómodamente desde su casa y frente al televisor. Se juega por teclado o con Joystick.

1906 Super Allien 4.500
Juego en cartucho. El individuo que usted maneja deberá construir defensas en el laberinto frente al ataque de sus enemigos que le pretenderán devorar.

1907 Júpiter Lander 4.500
Este juego en cartucho le permitirá demostrar su habilidad al intentar aterrizar en Júpiter en condiciones muy difíciles y con el peligro de que el combustible se le agote. Se juega por teclado.

1908 Draw Poker 4.500
Si quiere jugar al poker contra un enemigo implacable e impasible, este cartucho se lo permitirá. Se juega por teclado.

1909 Road Race 4.500
Con este cartucho podrá usted conducir un coche a través de una autopista. Se juega por cartucho.

1910 Radar Ratrace 4.500
Conduce un ratón a través de un laberinto procurando comer el queso y evitar que sea cazado por los gatos o por otros ratones que están al acecho. Mediante un radar sabe en qué posición se encuentran tanto sus enemigos como su objetivo: el queso. Se juega con Joystick o teclado.

1911 The sky is failling 4.500
Tiene que evitar a toda costa que el muro alcance su altura. Se juega con paddle.

1912 Mole attack 4.500
Debe darle con un martillo a un topo que va apareciendo por agujeros diversos. Sumará puntos positivos o negativos en función de si le da en la cabeza o en la base posterior.

1919 Sargon II chess 4.500
Juego de ajedrez con 7 niveles de dificultad. A partir del nivel 1 existe la opción de que el VIC-20 le indique qué jugada haría si estuviese en su lugar, por lo que le hace adecuado para los que se inicializan en este juego. Se juega por teclado o Joystick.

1924 Omega Race 4.500
Con este cartucho podrá pasar un rato divertido simulando batallas espaciales. Se juega con joystick.

C-400 VIC Forth . . . . consultar
Lenguaje de alto nivel.

C-401 VIC Stat . . . . consultar
Paquete de utilidades estadísticas.

C-402 VIC Graph . . . . consultar
Resolución de funciones por métodos gráficos.

C-403 Wordcraft 42.500
Compatible Series CBM 8000 y CBM 4000. Tratamiento de textos.

C-404 VIC Screen Master 10.000
Paquete de utilidades para edición de pantallas con utilidades para manejo de cadenas y disco.

BIBLIOGRAFÍA

Manual usuario VIC-20 800
Nociones de funcionamiento del VIC-20 y del lenguaje BASIC.

Curso Introducción BASIC: Parte I 2.500
De una manera clara y sencilla, partiendo de cero, se va enseñando el lenguaje del futuro. Este libro va acompañado de dos cassettes con 17 programas en castellano.

Curso Introducción BASIC: Parte II 2.500
Profundización de lo tratado en Parte I, especialmente en creación de ficheros, manejo de subrutinas, efectos musicales, aspectos animados, etc.

VIC Revealed 1.500
Libro que explica con detalle el funcionamiento interno del VIC-20. Incluye esquemas. Libro fotocopiado en inglés.

Programers Reference Guide 2.000
Profundización en el manejo del VIC-20. Incluye referencias a rutinas lenguaje máquina. Libro fotocopiado en inglés.

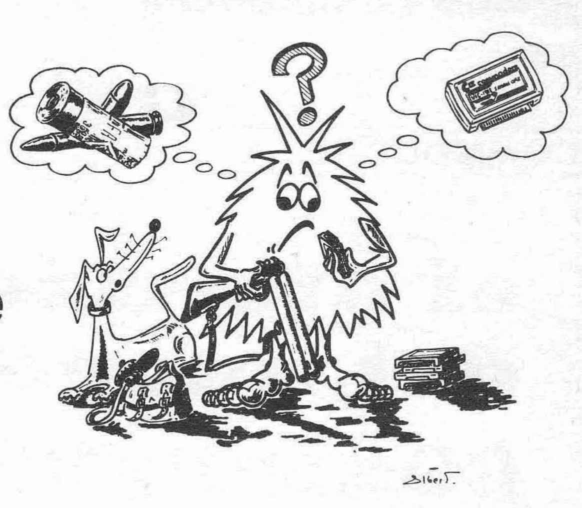
PROGRAMAS COMERCIALIZADOS POR "MICROELECTRÓNICA Y CONTROL, S.A."
Puede encontrarlos en su distribuidor más próximo

microelectrónica y control s.a





TRUCOS (I)



el cartucho de ayuda al programador

por R. PARDO

Como ya sabéis, la función AUTO tiene como objetivo asignar números de línea automáticamente en incrementos dados. Ahora bien, a la hora de copiar programas, no siempre éstos están numerados de una manera simétrica. Así que se hace preciso renunciar al modo AUTO (lo que es fastidioso) o bien definir muchos AUTOs (más fastidioso todavía). Por eso creo que es de especial interés este pequeño truco:

Supongamos que tenemos este programa:

```
10 PRINT «CLR»
15 PRINT «VIVA»
17 PRINT «YO»
20 PRINT «Y»
25 PRINT «MIS AMIGOS»
```

Si no fuera por la línea 17 podríamos aplicar AUTO 10,5 y el problema quedaría resuelto.

Apliquemos AUTO 10,5 y entremos todas las líneas hasta la 15. Entremos ésta y pulsemos RETURN. Aparecerá el número 20 (lógico) pero ahora vamos a desplazar el cursor hacia la izquierda, sobre el número de línea y teclearemos 17, entrando después la línea correspondiente y pulsando RETURN. ¿Qué ha pasado? ¡Nos aparece el número 20 otra vez!, y si sigue entrando el programa verá que seguirá respetando la numeración preestablecida. Si se usa este truco varias veces seguidas es posible que cambie el número de línea siguiente.

Otro truco útil es la asignación de funciones a las teclas programables del VIC. En el manual del cartucho dice que, para asignar funciones, deberá hacer KEYn, «xxx» donde la **n** es el número de tecla (1 a 12), y **«xxx»** es un texto que saldrá en pantalla al ser pulsada dicha tecla; puede ser indistintamente una palabra BASIC o no. Si quiere asociar al texto el retorno de carro (el equivalente a pulsar RETURN) el formato será el siguiente:

```
KEY,«xxx»+CHR$(13)
```

Donde CHR$(13) es el código de RETURN. Ahora bien, esto es largo de escribir y difícil de recordar. Así pues y sólo para los propietarios del cartucho «Programmer's Aid», hay un truco para hacer la misma función. ¿Habéis probado alguna vez el listar las funciones asignadas en las teclas de función? Si no lo habéis hecho nunca, teclead KEY y pulsad RETURN. Fijaos, en la lista, que las teclas que contienen el retorno de carro son F3 (RUN), F11 (RETURN), etc.

¿Qué veis en esas teclas? Si os fijais, el comando BASIC (p. e. RUN) va seguido de una flecha a la izquierda con inversión de color.

¿Habéis adivinado lo que es? Es la manera que tiene el cartucho de identificar el retorno de carro. ¿Qué pasará si introducimos ese nuevo carácter al final de una cadena alfanumérica de una tecla de función? (De paso diré que ese carácter se genera pulsando simultáneamente CTRL y BLUE.) Pues pasará lo que esperamos que pasaría. ¡Genera el retorno de carro!

Hablando del comando KEY diré que sirve para algo más que para listar las funciones asignadas: sirve también para volver al programa del cartucho cuando hemos salido de él, ya sea por haber pulsado RUN/STOP y RESTORE o por otra causa. Ahora bien, no lo devuelve después de un comando KILL.

Y de este comando vamos a hablar ahora, en relación a las funciones asignables que hemos visto hace un momento. Si hacéis KILL, anularéis el cartucho pero NO del todo. ¡Ojo!, pues esto puede ser una fuente de problemas. Las teclas de función quedan asignadas «in eternum» con las funciones que asignó el cartucho o asignasteis vosotros. Ni siquiera con el modo Reset del VIC (pulsando simultáneamente RUN/STOP y RESTORE) se «limpiarán» las funciones asignadas. Pero, ¡no desesperéis!, haciendo SYS 64802 lo limpiaréis. Ahora bien, como dice el refrán «es peor el remedio que la enfermedad» ya que esto inicializa del todo al ordenador y se «carga» el programa y todas las variables que hubiera en memoria en ese momento (como si hubiésemos desconectado el VIC y lo hubiéramos conectado de nuevo. ■

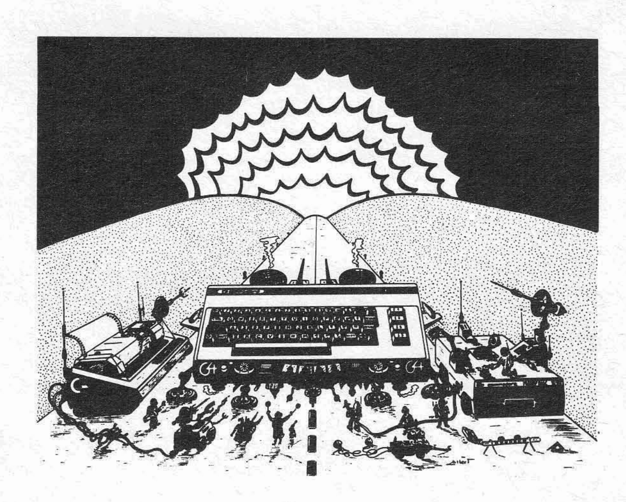
MINI-NOTAS

utilización de las teclas SHIFT y RUN/STOP

Si se pulsan las teclas SHIFT y RUN/STOP simultáneamente, el VIC cargará el primer programa —del cassette— que encuentre, y una vez cargado LO EJECUTARÁ AUTOMÁTICAMENTE.

COMMODORE 64

el aterrizaje del nuevo COMMODORE 64



por **P. MASATS**

En este artículo vamos a intentar describir — brevemente — los cuatro circuitos integrados sobre los que COMMODORE ha construido su nuevo modelo C-64 y que constituyen, a nuestro entender, las cuatro piezas fundamentales de su «tren de aterrizaje».

En estos tiempos estamos acostumbrados a las maravillas realizadas por los programas en los ordenadores personales, olvidando fácilmente que el Software por sí solo poco puede hacer sin unos cimientos sólidos de Hardware. En este sentido, COMMODORE, apoyándose en su empresa filial MOS TECHNOLOGY — que no debe olvidarse que fue la que desarrolló el microprocesador 6502 — ha dotado al COMMODORE 64 de cuatro magníficas «patas» que le unen firmemente al terreno que pisa.

Estos cuatro circuitos integrados son los siguientes:

MOS 6510 CPU
(unidad central de proceso).
MOS 6566 VIC-II
(controlador de video).
MOS 6581 SID
(sintetizador de sonido).
MOS 6526 CIA
(controlador de periféricos).

Veamos ahora sus posibilidades de uno en uno:

MOS 6510. — CPU en tecnología NMOS y encapsulado de 40 patillas D.I.L. Es igual al clásico MOS 6502 que equipa no sólo a los productos COMMODORE sino a muchos otros en el mercado de ocho bits. La única diferencia consiste en que posee un port de entrada/salida de ocho bits en la posición de memoria $0000 cuya dirección de intercambio de datos se determina por el contenido de la dirección $0001. La función principal de este registro de 8 bits es controlar la configuración de memoria interna del C-64, permitiendo el funcionamiento simultáneo de los 64 Kbytes de RAM (seguro que algún lector debía pensar que nos olvidábamos de los 64 K de RAM) y de los programas en ROM (el sistema operativo, el intérprete de BASIC, el generador de caracteres y los cartuchos opcionales de ROM que puedan conectarse). En relación con la memoria direccionada por la CPU diremos que la parte de RAM está formada por ocho chips 4164-2 de 65536 × 1 bit de memoria dinámica de cuyo refresco se encarga el controlador de video (MOS 6566), lo cual simplifica considerablemente el circuito y, en consecuencia, minimiza los riesgos de avería. Esta memoria RAM cubre todo el espacio direccionable por la CPU. Sin embargo, en condiciones normales, algunas secciones de memoria RAM se pueden sustituir por secciones en ROM. Estas secciones pueden ser: KERNAL o sistema operativo, ocupa de $E000 a $FFFF (8K) y mediante la acción de un bit del registro de la CPU puede ser sustituido por RAM. De $D000 a $DFFF (4K) existe un espacio que puede contener el generador de caracteres, memoria RAM o una región destinada a entradas/salidas. De $A000 a $BFFF (8K) se sitúa el intérprete de BASIC que también puede ser eliminado y sustituirse por RAM, lo cual facilita el trabajo con otros lenguajes diferentes del BASIC. Y para terminar, entre $8000 y $9FFF se puede situar un cartucho exterior de 8K de ROM o, como siempre, se puede disponer de la RAM que hay «debajo». (Nótese que con el C-64 empezamos a trabajar con mapas de memoria tridimensionales. ¡Que BUG nos asista a los que tenemos intención de llegar a viejos en la microinformática!) A todos los efectos, el resto de memoria es siempre RAM, lo cual nos deja libres para BASIC ¡38911 Bytes! y 54 K en ASSEMBLER.

MOS 6566 VIC-II (Video Interface Chip II ó Chip de interface de video de la segunda generación). — Este circuito integrado de 40 patillas D.I.L. en tecnología NMOS va a ser, sin ninguna duda, el culpable de más de un ataque de nervios en la Redacción de nuestra Revista, pues presenta al procesador nada menos que 47 registros para manejar sus parámetros de funcionamiento. Permite — simultánea o separadamente — tres tipos de imágenes: SPRITES o Bloques y Objetos Móviles, el juego de caracteres normal de COMMODORE y Gráficos en alta resolución. Los SPRITES son ocho bloques de 24 por 21 puntos que se pueden definir independientemente punto por punto y que se pueden mover por la pantalla en cualquier dirección y pasar — desde el punto de vista del observador — por detrás de otro SPRITE según un orden de procedencia determinado. Cada SPRITE puede exhibirse con uno de los dieciséis colores con que puede trabajar el C-64 en modo normal o hasta cuatro colores diferentes (incluyendo el del fondo de la pantalla), reduciéndose su definición a la mitad (cada SPRITE tendría 12 × 21 puntos). Además cada SPRITE puede ampliarse en sentido horizontal y/o vertical por separado y mediante registros específicos se puede detectar su «colisión» con otro SPRITE o gráfico de otro tipo presente en la pantalla.

El funcionamiento en modo caracteres es muy similar al del VIC-20 ex-

cepto que se dispone en el C-64 de 16 colores. Los 8 primeros se obtienen del mismo modo que en el VIC y los 8 restantes pulsando la tecla COMMODORE y una de las ocho teclas de color. El juego de caracteres se puede definir por el usuario y, dada la disponibilidad de memoria RAM, esto se realiza con gran facilidad. La pantalla tiene 40 columnas y 25 líneas. El modo caracteres permite dos submodos: el multicolor y el extendido.

En modo alta resolución, cada bit de una región de memoria específica se convierte en un punto en la pantalla, permitiendo realizar cualquier gráfico con una resolución de 320 por 200 puntos en dos colores, pudiéndose trabajar con cuatro colores sacrificando la resolución que pasaría a ser de 160 por 200.

Por si todo esto fuera poco, el MOS 6566 tiene una serie de posibilidades de tipo general. El manejo de un solo bit en un determinado registro puede permitir cosas tales como: borrado de la pantalla y liberación del BUS del procesador por parte del controlador de video (se admiten apuestas sobre quién ejerce más control sobre el BUS, el controlador de video o la CPU); reducción del formato de pantalla a 24 líneas de 38 caracteres (muy útil para deslizamientos) y RESET (inhabilitación) total del controlador. Por último (estamos casi convencidos de no haber olvidado nada), es posible deslizar (SCROLL en inglés) la información en pantalla punto a punto, tanto en sentido vertical como horizontal. Para ello es muy útil la reducción de pantalla para tener un lugar donde crear la información que luego se ha de deslizar a lo largo (o a lo ancho) de la misma.

MOS 6581 SID (Sound Interface Device, o dispositivo de interface de sonido). — Éste es un circuito integrado de 28 patillas D.I.L. en tecnología NMOS y quizá la más innovadora de las recientes creaciones de COMMODORE-MOS TECHNOLOGY. En este circuito integrado está contenido un completo sintetizador musical del tipo MOOG con la salvedad de que, en vez de controlar su funcionamiento por tensiones entre los diferentes módulos, se utilizan aquí números binarios de ocho o más bits. En la siguiente lista se dan las características de los elementos que componen este circuito y entre paréntesis las denominaciones equivalentes que se utilizan en la elaboración de música con sintetizador. Los elementos que componen el SID son:

— Tres generadores de señal (VCO y LFO) con control de frecuencia entre 0 y 4 KHz. (8 octavas desde el DO de 16.35 Hz al SI de 3951.06 Hz de la escala temperada). Estos osciladores pueden generar señales triangulares, diente de sierra, cuadrada con ancho de pulso variable y ruido.
— Tres generadores de envolvente (ADSR) de respuesta exponencial, asociados a cada VCO y con controles independientes de tiempos de Ataque, Decaimiento, Relajación y de nivel de Sostenimiento.
— Tres moduladores de amplitud (VCA) con una gama dinámica de 48 dB enlazan los ADRS y los VCO. Se incluye posibilidad de sincronización de los VCO y modulación en anillo.
— Filtro programable (VCF) de 12 dB/octava con frecuencia de resonancia variable de 30 Hz a 12 KHz. Es configurable como pasa banda, Banda eliminada, pasa alto y pasa bajo. También se puede controlar la resonancia del filtro. Se incluye un control general de volumen, dos entradas para potenciómetros y una entrada para señal de audio exterior que puede ser procesada por el filtro.

MOS 6526 CIA (Complex Interface Adapter, o Adaptador Complejo de Interface). — En tecnología NMOS este circuito integrado de 40 patillas D.I.L. constituye la tercera generación de controladores flexibles de periféricos, siendo las dos primeras la MOS 6520 PIA y la MOS 6522 VIA. Sus características son:

— 16 líneas de entrada/salida programables individualmente.
— Transferencia de información con pautas de acuerdo (HANDSHAKING) en base a 8 ó 16 bits en lectura o escritura.
— Dos registros de intervalos de tiempo independientes y encadenables de 16 bits.
— Reloj interno de 24 horas (AM/PM) con alarma programable.
— Registro de desplazamiento de 8 bits para E/S serie.
— Capacidad de dos cargas TTL.
— Líneas de E/S compatibles con CMOS.

En el C-64 se incluyen dos de estos chips que permiten el manejo del teclado y otros periféricos incluyendo el port del usuario con el interface RS232.

De momento, esto es lo que sabemos del nuevo COMMODORE 64 y aunque difícilmente estos datos pueden dar una idea de lo que realmente puede hacer este Ordenador, creemos que queda plenamente justificada la frase: EL COMMODORE 64, EL MEJOR ORDENADOR PERSONAL DEL MUNDO. Seguiremos informando. ■

MULTICENTRO DE INFORMATICA

Chips & Tips

PUERTO RICO, 21-23 - MADRID-16 TEL. 250 74 02 - 250 74 04

commodore VIC-20

JUEGOS

- COMECOCOS. 3,5K. A/R. G/E. JY. EXTRAORDINARIA VERSION DEL POPULAR PUCKMAN. COLOR Y MOVILIDAD INSUPERABLE . 1.900
- VICGAMON. +3K. JUEGO DE INTELIGENCIA QUE LE MANTENDRA EN TENSION HASTA DERROTAR A SU VIC 1.800
- ASTEROIDS WAR. 3,5K. C/M. A/R. G/E. JY. ESPECTACULAR BATALLA GALACTICA CONTRA LA NUBE PROTONICA EN 3 DIMENSIONES 1.800
- FROGGER. +3K y 3,5K. C/M. A/R. G/E. JY. ULTIMA NOVEDAD EN EE.UU. CRUZAR EL PELIGROSO RIO Y LA AUTOPISTA SUICIDA 2.000
- RATMAN. +8K. C/M. A/R. G/E. DE LA BOVEDA CELESTE DESCENDERAN EXTRAÑAS RATAS ATOMICAS. ESPECTACULAR ANIMACION 1.900
- SHARK ATTACK. 3,5K. C/M. A/R. JY. EN MEDIO DEL OCEANO SERA ATACADO POR LOS PELIGROSOS TIBURONES. DEFIENDASE CON SU RED ATOMICA.................. 1.900
- ROX III. 3,5K Y+8K. C/M.A/R. G/E. JY. DESDE SU SOFISTICADA BASE LUNAR DEFIENDA SU PLANETA DEL ATAQUE DE LOS UFOS 1.800
- ,ULTISOUND SYNTHETIZER. 3,5K. ¿UN ORGANO EN SU VIC? ¿CON ACOMPAÑAMIENTO, BATERIA Y EFECTOS ESPECIALES? .. 1.900
- SKI-RUN. 3,5K. C/M. A/R. G/E. DESLICESE POR LAS HELADAS PISTAS DE COMPETICION. SLALOM, S/GIGANTE, DESCENSO. 9 NIVELES 1.800
- FIREBIRD. (SPACE PHREEKS). 3,5K. C/M. A/R. G/E. JY. AÑO 3.010. VD. ES EL UNICO SUPERVIVIENTE DE LA BATALLA DE RIGELLIAN. DEBERA COLONIZAR OTRO PLANETA Y LUCHAR CONTRA LAS CRIATURAS GALACTICAS 1.900
- BREAKOUT. 3,5K. CONSIGA DESTRUIR LA PARED DE LADRILLOS MULTICOLORES CON LA BOLA MAGICA. INCLUYE "MASTERMIND". 1.600
- AJEDREZ. PRIMERA VERSION EN CASSETTE CON GRAFICOS EN ALTA RESOLUCION. BASTANTES NIVELES DE JUEGO. (STANDARD) . 2.800
- SHADOWFAX. INCREIBLES GRAFICOS ANIMADOS. EL CABALLERO DE LAS SOMBRAS EN LUCHA CONTRA LOS JINETES DEL TIRANO INVASOR. (STANDARD)................ 1.900
- SNAKE. COLORIDO, MOVIMIENTOS Y GRAFICOS EXCEPCIONALES. VERSION DEL FAMOSO JUEGO DE LAS SERPIENTES (SNAKE). (STANDARD) 1.900
- MYRIAD. +3K. C/M. A/R. G/E. JY. LA MAS ESPECTACULAR AERONAVE PARA DESTRUIR LAS CRIATURAS COSMICAS EN SU VIAJE GALACTICO................ 2.000
- COSMIADS. 3,5K. C/M. A/R. G/E. JY. VERSION ULTRARRAPIDA DEL MUNDIALMENTE FAMOSO JUEGO "GALAXIANS". INCREIBLES EFECTOS SONOROS................ 1.700
- BLITZRIEG. 3,5K. C/M. A/R. G/E. JY. DESTRUYA LA CIUDAD ENEMIGA CON SU BOMBARDERO. 25 NIVELES DE JUEGO 1.600
- DEFENSA. +8K. C/M. A/R. G/E. JY. N.º 1 EN INGLATERRA, COMO GUERRERO GALACTICO DEBE DEFENDER A LOS HUMANOIDES CONTRA LOS ENEMIGOS CIBERNETICOS. 9 NIVELES DE JUEGO. ESPECTACULAR NAVE Y SONIDOS 2.000
- VIC PANIC. 3,5K. C/M. A/R. G/E. JY. VERSION DEL POPULAR "SPACE PANIC". ESCALE LAS LADERAS Y HUYA DE LOS MONSTRUOS 1.800
- SKRAMBLE. 3,5K. C/M. A/R. G/E. JY. ATRAVESANDO LOS TEMIBLES PASADIZOS INTERESTELARES DESTRUYA LAS BASES ENEMIGAS 1.900
- 3D LABYRINTH. +8K. C/M. A/R. EXTRAORDINARIO LABERINTO TRIDIMENSIONAL. ¿SERA CAPAZ DE SALIR DE EL? UNO O VARIOS JUGADORES 1.800
- GOLF. 3,5K. RECORRIDO DE 9 HOYOS PERO ATENCION A LOS OBSTACULOS: ARBOLES, LAGOS, ETC. INCLUYE VIC MUSIC Y PIANO 1.600
- CARRERA DE BUGGYS. 3,5K. C/M. A/R. G/E. ESPECTACULAR RECORRIDO. ACELERADOR. DECELERACION. 9 NIVELES 1.800
- GRIDRUNNER. 3,5K. C/M. A/R. G/E. JY. IMPRESIONANTE VERSION LLENA DE COLORIDO, MOVILIDAD Y SONIDO DEL POPULAR "CENTIPEDE"................ 1.900
- HI-RES. 3,5K. GRAN JUEGO DEMO/UTILIDAD PARA REALIZAR EN PANTALLA GRAFICOS EN ALTA RESOLUCION. INCLUYE GEN. CARACTERES................ 1.500
- ABDUCTOR. LAS CRIATURAS COSMICAS DEL PLANETA "ALPHA I" INTENTARAN SECUESTRAR A LOS HUMANOIDES PARA CONSEGUIR ENERGIA E INTELIGENCIA SUPERIORES. TU MISION SERA DEFENDER TU PLANETA Y DESTRUIR LAS NAVES ABDUCTORAS. (STANDARD). 1.800
- TRAXX. VERSION DEL CONOCIDO JUEGO "AMIDAR"; MEZCLA DEL POPULAR "PACKMAN" Y DEL JUEGO "QUIX". 100% CODIGO MAQUINA. GRAFICOS EN ALTA RESOLUCION. ESPECTACULAR SONIDO Y COLOR. 8K DE MEMORIA 2.000

UTILIDADES

- VIC PRINT. +8K. EXTRAORDINARIO Y SENCILLO PROCESADOR DE TEXTOS. TABULACION, MAQUETACION, CABECERAS, COPIAS. CASS O DISK................ 2.000
- VIC LABEL. +8K. EN COMBINACION CON VIC PRINT, ELABORA ETIQUETAS PARA DIRECCIONES................ 1.900
- VIC POST. +8K. ELABORA LETRAS Y TEXTOS ESPECIALES EN TAMAÑO Y FORMA PARA POSTERS, LISTAS DE PRECIOS, ETC. .. 2.900
- VIC CALC. HERRAMIENTA DE CALCULO QUE SUSTITUYE AL LAPIZ, PAPEL Y CALCULADORA, REALIZA COMPLEJOS MODELOS FINANCIEROS CON POSIBILIDAD DE AJUSTARLO A OTROS PARAMETROS CON SOLO PULSAR UNA TECLA. 16K DE MEMORIA............ 3.200
- VIC BASE. 16K. POTENTE BANCO DE DATOS. 255 CARACTERES, MAS DE 25 CAMPOS. CAMBIO Y LOCALIZACION, SALIDA IMPRESORA................ 3.200
  OPCION 3,5K................ 1.800
- GRAPHVICS. +3K. AÑADE 18 POTENTES COMANDOS PARA POSICIONAR PUNTOS, DIBUJAR LINEAS Y TEXTOS EN ALTA RESOLUCION (152×160) 2.200
- GRAPH EDITOR & SOFTKEY 24, 3,5K. AMBOS PROGRAMAS PERMITEN DISEÑAR HASTA 64 CARACTERES PARA INCORPORARLOS A SUS PROPIOS PROGRAMAS Y JUEGOS 2.000

EDUCATIVOS

- QUIZ-MASTER. +3K. EL MAS ESPECTACULAR AVANCE EDUCATIVO. PERMITE LA CORRECCION Y PUNTUACION DE TODAS LAS RESPUESTAS QUE RECIBE EL ORDENADOR................
- QUIZ SET-UP. EN TANDEM CON QUIZ—MASTER PERMITE LA ELABORACION POR EL USUARIO DE TODO TIPO DE PREGUNTAS Y CUESTIONES EDUCATIVAS O DE ENTRETENIMIENTO, EGB, IDIOMAS, MATEMATICAS, HISTORIA, GEOGRAFIA, ETC. CREANDO UN AGIL Y ATRACTIVO SISTEMA DOMESTICO/EDUCATIVO .....
  (QUIZ-MASTER y QUIZ SET-UP) 3.200
- FACEMAKER. 16K. CARICATURANDO EL ROSTRO DE SUS COMPAÑEROS Y AMIGOS EL VIC 20 PONDRA A PRUEBA EL VOCABULARIO Y LA ATENCION DEL NIÑO................ 2.000
- NUMBER CHASER. 16K. PROGRAMA PARA PRACTICAS DE MULTIPLICACION CON CARRERAS DE COCHES, ADELANTA, FRENA, ACELERA SEGUN LAS RESPUESTAS. 4 NIVELES DE DIFICULTAD ... 2.000
- NUMBER GULPER. 16K. JUEGO EDUCACIONAL DE COMPETICION CON NUMEROS PARA SUMA, RESTA, MULTIPLICACION Y DIVISION 2.000
- WE WANT TO COUNT. 16K PROGRAMA PARA NIÑOS A PARTIR DE TRES AÑOS, INVASORES, CARRERAS, ETC. 2.000
- TWISTER. 16K. JUEGO DE LOGICA Y CONCENTRACION. PUZZLES GEOMETRICOS CON SONIDO Y COLOR................ 2.000

LIBROS

VIC REVEALED	2.200	ASSEMBLER	2.000	ZAP! POW! BOOM!	1.800
GETTIN ACQUAINTED WITH YOUR VIC 20	1.800	SYNPHONY MELANCHOLY COMP.	1.800	VIC INNOVATIVE	2.000
50 PROGRAMAS LISTADOS I	1.500	50 PROGRAMAS LISTADOS II	1.500	50 PROGRAMAS LISTADOS III	1.500

NOTICIAS

¡última hora: acaba de aparecer la segunda parte del curso de BASIC!

por **P. MASATS**

Después de ser esperada por más de un entusiasta que no ha dejado de hacernos constar su impaciencia, ya se ha puesto a la venta el libro y las cintas que constituyen esta segunda parte del curso de BASIC que tanto éxito ha tenido. Como pensamos que valen más hechos que palabras, en la figura 1 se puede ver el contenido de las diferentes unidades didácticas que componen este curso.

Como CLUB COMMODORE tiene una activa plantilla de espías, éstos han conseguido un programa de los que se incluyen en los cassettes de este curso para su publicación en la Revista. No obstante, para que no se pueda decir que somos unos copiones, lo hemos modificado para nuestros lectores.

El programa cuyo listado se puede ver en la figura 2 nos permite ejecutar en el VIC-20 una partitura escrita para piano u otro instrumento de teclado (en este caso el clave). Hasta la línea 500 inclusive, se trata del «sistema operativo» que se encarga de transformar las declaraciones DATA, contenidas en las líneas 1000 en adelante, en la canción que debe interpretarse. En la línea 1, la variable V establece la velocidad de ejecución.

En una partitura para clave (o piano) como la que se puede ver en la figura 3 y que ha servido de base para la confección del programa de la figura 2, se pueden observar dos conjuntos de cinco líneas paralelas o pentagramas que corresponden a las notas que se deberán ejecutar con cada mano; el pentagrama superior es para la mano derecha y el inferior para la izquierda. En una partitura, la forma de la nota designa su duración y su

(continúa en la pág. siguiente)

```
READY.

1 V=4

2 PRINT"[ CLR ]MUSSETTE

3 PRINT"[CRSRD]DEL LIBRO DE CLAVE DE
ANA MAGDALENA"

4 PRINT"[CRSRD]DE JUAN SEBASTIAN BACH"

5 FORJ=36874 TO 36878:POKE J,0:NEXT J

10 DATA0,131,138,145,151,157,162,167,172
,177,181,185,189

20 DATA193,196,200,203,206,209,211,214,2
16,218,220,222,224,226,228,229,231

30 DIMZ(29)

40 FORJ=0TO29:READZ(J):NEXTJ

50 POKE36878,10

60 READA$,B$,C$,D$

70 IFA$="Z"THEN 500

80 P=1:Q=1:R= 0:S=R:L=0

90 L=L+1 :IFL <RTHEN140

100 IFLEN(MID$(A$,P,1))=0THEN 60

110 N=ASC(MID$(A$,P,1))-64

120 R=L+V*(ASC(MID$(B$,P,1))-48)

130 G=Z(N):P=P+1

140 IF L <STHEN190

150 N=ASC (MID$(C$,Q,1))-40

160 S=L +V*(ASC(MID$(D$,Q,1))-48):Q=Q+1

165 IFN=0THENH=0:I=0:GOTO190

170 IFN<13THENI=Z(N+12):H=0 :GOTO190

180 I=0:H=Z(N)

190 POKE36876,G:POKE36875,H:POKE36874,I:
GOTO90


250 IFL>=STHEN150

280 GOTO90

500 POKE36878,0:END

1000 DATA"VTSQOVTSQOGHJHGEJGCVTSQOVTSQOG
HJHGEJC"

1010 DATA"41111411111222222241111411111
1222224"

1020 DATA"+7+7+7+7;<><;9>;7+7+7+7+7;<><;
9>7"

1030 DATA"222222221122222222222222211222
224"

1040 DATA"VTSQOVTSQOGHJHGEJGCVTSQOVTSQOG
HJHGEJC"

1050 DATA"41111411111222222241111411111
1222224"

1060 DATA"+7+7+7+7;<><;9>;7+7+7+7+7;<><;
9>7"

1070 DATA"222222221122222222222222211222
224"

1080 DATA"NOQNOQVQVQVQONLJLEQPEOONVU"

1090 DATA"1121122622221111222222222"

1100 DATA"2>2>2>2>2>2>2>-9-9-9-9-9"

1110 DATA"2222222222222222222222222"

1120 DATA"QPEOONVUQPNPQPNPQIJONOQJCBCQJ"

1130 DATA"222222221111111222211222112 4"

1140 DATA"-9-9-9-9-9-9-9679276792"

1150 DATA"22222222222222224221124"

1160 DATA"VTSQOVTSQOGHJHGEJGCVTSQOVTSQOG
HJHGEJC"

1170 DATA"41111411111222222241111411111
1222228"

1180 DATA"+7+7+7+7;<><;9>;7+7+7+7+7;<><;
9>7"

1190 DATA"222222221122222222222222211222
228"

2000 DATAZ,Z,Z,Z

READY.
```

Fig. 2

acaba de aparecer la segunda parte del curso de BASIC

(continuación)

situación en el pentagrama nos dice qué nota es. En nuestro programa la serie de notas que constituyen la pieza musical se dividen en cuatro series de cadenas alfanuméricas que contienen sucesivamente las notas de la mano derecha (línea 1000), las duraciones correspondientes a dichas notas (línea 1010), las notas de la mano izquierda (línea 1020) y sus duraciones (línea 1030) según el código que se da en la figura 4. En la figura 5 se da la extensión (las notas extremas) que se cubre por parte del programa para cada «mano». En las líneas mencionadas se ejecuta la partitura hasta la barra vertical más gruesa que se puede ver casi al principio de la segunda línea de la partitura. Esta línea

(sigue en la pág. 20)

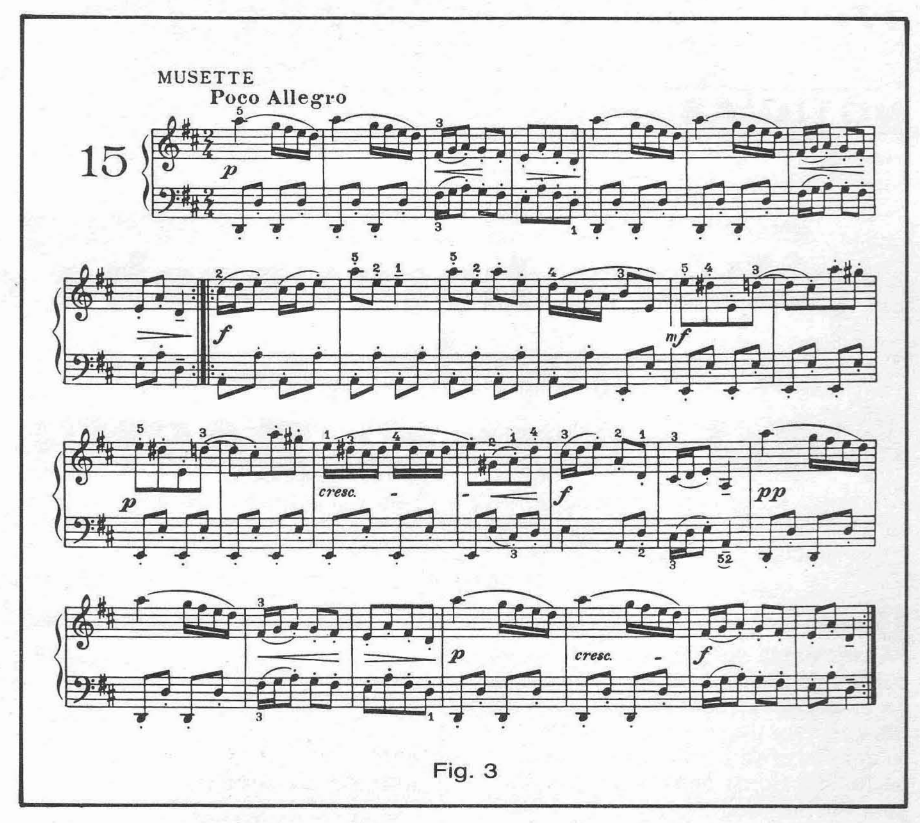


Fig. 3

TABLA DE MATERIAS o (INDICE DE MATERIAS)

Título	Temática	Programas grabados en el cassette	Pag.
	Introducción		
Unidad 16	Las instrucciones DATA, READ y RESTORE Sirviéndose de un bucle; las instrucciones DATA y READ; Formato de una instrucción DATA; errores en las instrucciones DATA y READ. El comando RESTORE.	UNI16CUEST	153
Unidad 17	Tratando con problemas complejos; el uso de los dos puntos; Usos de las instrucciones IF-THEN; Operadores lógicos; El operador AND; el Operador «OR»; Combinación de operadores lógicos; el comando NOT.	UNI17PROG	161
Unidad 18	Introducción a las subrutinas; Formato de las subrutinas; Como funciona GOSUB; transmisión de parámetros desde/a una subrutina.	CUADRO	171
Unidad 19	Más detalles sobre las subrutinas; Las especificaciones de subrutina; subrutina de simplificación de fracciones; Fiabilidad de una subrutina; Limitaciones a la capacidad de un parámetro; Convenciones en la denominación de subrutinas.	GRANLETRAS	181
Unidad 20	Matrices: Las instrucciones de DIMENSION; El uso de las variables de matriz; El uso «avanzado» de matrices. DATA.	UNI20CUEST	191
Unidad 21	Funciones de cadenas; La función LEN; La función MID$: La extracción del apellido; Uso de MID$ para rectificar una cadena; Funciones LEFT$ Y RIGHT$; Permutaciones; Eliminación de letras de una cadena Conversión de una cadena en números: VAL; Como evitar que se corten palabras en la pantalla	UNI21CUEST	199
Unidad 22	Uso de matrices para búsqueda y clasificación; El "corte binario"; Método de clasificación "BUBBLESORT"; Método rápido de clasificación; Comparación de tiempos de clasificación; La función FRE; Agotamiento del espacio de Memoria; Matrices bidimensionales.	CLASRAP, EMPIEZA VIDA	215
Unidad 23	Una ojeada al interior del VIC; Organización de Memoria del VIC; BYTES; El comando PEEK; El comando Poke; Introduciendo la animación; Más detalles sobre PEEK y POKE; Ejemplo de animación.	AVISPAS	229
Unidad 24	Más detalles sobre operadores lógicos; Como se evaluan condiciones con el VIC; Códigos ASCII de CBM; La función ASC; El comando ON; El comando END; El comando DEF; Almacenaje y recuperación de datos en cassette; El comando PRINT; El comando INPUT; El comando GET	HACE NOMBRES	247
Unidad 25	Diseño de Programas—Estudio de algunos casos; Frases aleatorias; Juegos de aventuras o laberintos.	RESCESP, CASTILLO	263
Conclusión			
Apéndice A	Interpretación de Música con el VIC	PRELUDIO VIBRATO GAVOTA TECLADO	279
Apéndice B	Biblioteca de Subrutinas	LIBRERIA	291
Apéndice C	Respuestas a experimentos		299
Indice alfabético			311

Fig. 1

acaba de aparecer la segunda parte del curso de BASIC

(viene de las págs. anteriores)

tiene, al lado, dos pequeños puntos que nos indican que llegados a este punto de la ejecución debe repetirse ésta desde el último signo del mismo tipo o desde el principio en caso de no encontrar ninguno. Así pues las líneas 1040 a 1070 son iguales a las 1000-1030. De hecho deberían repetirse por la misma razón las líneas 1080-1190, pero esto no se ha hecho así para no alargar excesivamente el listado. Para más detalles sobre este programa consultar (¡naturalmente!) el Apéndice **a** de la Parte II de nuestro Curso de Introducción al Lenguaje BASIC. ■
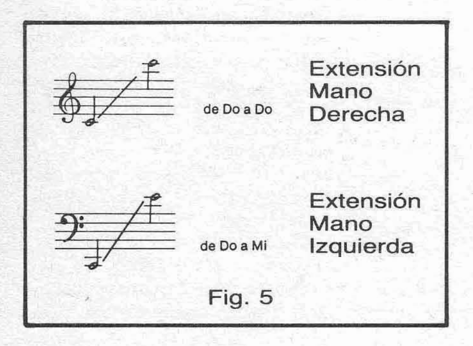


Fig. 5

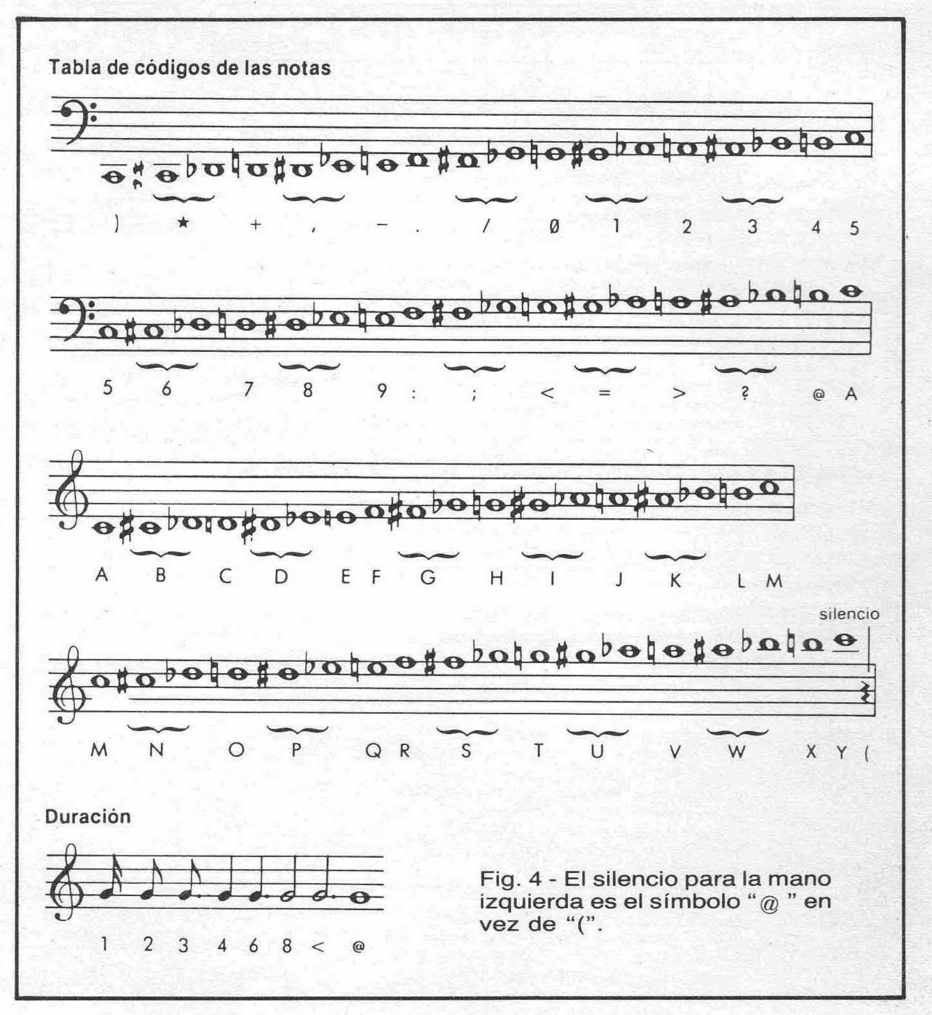


Fig. 4 - El silencio para la mano izquierda es el símbolo "@" en vez de "(".

MARKETCLUB

La posibilidad de realizar ventas, compras, intercambios, constituye un medio de intercomunicación entre los que comparten una misma afición, una misma actividad, idénticos afanes. Para fomentar en el seno del "Club Commodore" las relaciones entre sus miembros, mantiene abierta esta Sección en la que se da buena acogida a los textos de ofertas o peticiones relativas a los diferentes modelos de micro-ordenadores "Commodore", a sus periféricos, a programas, a libros, a información... Extensión máxima por comunicación: cincuenta palabras.

- **En Barcelona, clases de informática a particulares (horas convenidas) y grupos reducidos (sábados mañana). Lenguaje BASIC. Prácticas con micro-ordenador VIC-20. Prof. E. Martínez de Carvajal. Información: Tel. 345 10 00, Srta. María José (mañanas) ó 345 87 75 Sr. Martínez (fuera de horas de oficina).**

- **Vendo cartucho 16K VIC-20, por 14.000 ptas. Hago programas en Basic de Commodore (todas las versiones) bajo encargo. Desearía contactar con usuarios de Commodore en la zona de Madrid, para cambio de programas, impresiones, pokes especiales, etc... Razón: Francisco Gutiérrez. Santiago Rusiñol, 12. MADRID-3. Tel. (91) 253 13 40. Horas comida y cena.**

- **Vendo aplicación de facturación con control de representantes, 9 listados, 6 ficheros, estadísticas, etc., permite copias de seguridad. Configuración: VIC, 8K, disco e impresora, 40.000 pesetas. Escribir a Jaime Ameller Pons. General Mola, 15, 1.° B. CALATAYUD (Zaragoza).**

- **Se busca experto en VIC-20 para colaborar en la creación y coordinación de un Club de Usuarios de VIC en Barcelona. Llamar a Srta. Rosa Romero. Tel. 211 54 40.**

- **Se ofrece 3008 + C2N de segunda mano en buenas condiciones, precio a convenir. Razón: Domingo Garrofé Trabal. C/. Aragón, 386, 1.° - 1.ª. BARCELONA-9. Tel. 245 51 44.**

- **Vendo interfaz y programa para RTTY y CW para el PET a 15 K. Rafael, EA3CGK - Avda. Barcelona, 21, A, 4.° 2.ª. IGUALADA (Barcelona).**

Hex	Decimal	Description
00A0-00A2	160-162	Jiffy Clock HML
00A3	163	Serial bit count/EOI flag
00A4	164	Cycle count
00A5	165	Countdown,tape write/bit count
00A6	166	Tape buffer pointer
00A7	167	Tp Wrt ldr count/Rd pass/inbit
00A8	168	Tp Wrt new byte/Rd error/inbit cnt
00A9	169	Wrt start bit/Rd bit err/stbit
00AA	170	Tp Scan;Cnt;Ld;End/byte assy
00AB	171	Wr lead length/Rd checksum/parity
00AC-00AD	172-173	Pointer: tape bufr, scrolling
00AE-00AF	174-175	Tape end adds/End of program
00B0-00B1	176-177	Tape timing constants
00B2-00B3	178-179	Pntr: start of tape buffer
00B4	180	1=Tp timer enabled; bit count
00B5	181	Tp EOT/RS232 next bit to send
00B6	182	Read character error/outbyte buf
00B7	183	# characters in file name
00B8	184	Current logical file
00B9	185	Current secndy address
00BA	186	Current device
00BB-00BC	187-188	Pointer to file name
00BD	189	Wr shift word/Rd input char
00BE	190	# blocks remaining to Wr/Rd
00BF	191	Serial word buffer
00C0	192	Tape motor interlock
00C1-00C2	193-194	I/O start address
00C3-00C4	195-196	Kernel setup pointer
00C5	197	Last key pressed
00C6	198	# chars in keybd buffer
00C7	199	Screen reverse flag
00C8	200	End-of-line for input pointer
00C9-00CA	201-202	Input cursor log (row, column)
00CB	203	Which key: 64 if no key
00CC	204	0=flash cursor
00CD	205	Cursor timing countdown
00CE	206	Character under cursor
00CF	207	Cursor in blink phase
00D0	208	Input from screen/from keyboard
00D1-00D2	209-210	Pointer to screen line
00D3	211	Position of cursor on above line
00D4	212	0=direct cursor, else programmed
00D5	213	Current screen line length
00D6	214	Row where curosr lives
00D7	215	Last inkey/checksum/buffer
00D8	216	# of INSERTs outstanding
00D9-00F2	217-242	Screen line link table
00F3-00F4	243-244	Screen color pointer
00F5-00F6	245-246	Keyboard pointer
00F7-00F8	247-248	RS-232 Rcv pntr
00F9-00FA	249-250	RS-232 Tx pntr
00FF-010A	256-266	Floating to ASCII work area
0100-103E	256-318	Tape error log
0100-01FF	256-511	Processor stack area
0200-0258	512-600	Basic input buffer
0259-0262	601-610	Logical file table
0263-026C	611-620	Device # table
026D-0276	621-630	Sec Adds table
0277-0280	631-640	Keybd buffer
0281-0282	641-642	Start of Basic Memory
0283-0284	643-644	Top of Basic Memory
0285	645	Serial bus timeout flag
0286	646	Current color code
0287	647	Color under cursor
0288	648	Screen memory page
0289	649	Max size of keybd buffer
028A	650	Repeat all keys
028B	651	Repeat speed counter
028C	652	Repeat delay counter
028D	653	Keyboard Shift/Control flag
028E	654	Last shift pattern
028F-0290	655-656	Keyboard table setup pointer
0291	657	Keyboard shift mode
0292	658	0=scroll enable

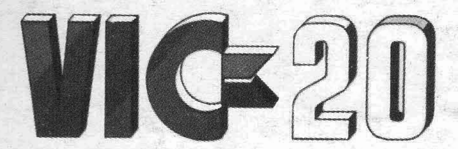
Hex	Decimal	Description	
0293	659	RS-232 control reg	
0294	660	RS-232 command reg	
0295-0296	661-662	Bit timing	
0297	663	RS-232 status	
0298	664	# bits to send	
0299-029A	665	RS-232 speed/code	
029B	667	RS232 receive pointer	
029C	668	RS232 input pointer	
029D	669	RS232 transmit pointer	
029E	670	RS232 output pointer	
029F-02A0	671-672	IRQ save during tape I/O	
02A1	673	CIA 2 (NMI) Interrupt Control	
02A2	674	CIA 1 Timer A control log	
02A3	675	CIA 1 Interrupt Log	
02A4	676	CIA 1 Timer A enabled flag	
02A5	677	Screen row marker	
02C0-02FE	704-766	(Sprite 11)	
0300-0301	768-769	Error message link	
0302-0303	770-771	Basic warm start link	
0304-0305	772-773	Crunch Basic tokens link	
0306-0307	774-775	Print tokens link	
0308-0309	776-777	Start new Basic code link	
030A-030B	778-779	Get arithmetic element link	
030C	780	SYS A-reg save	
030D	781	SYS X-reg save	
030E	782	SYS Y-reg save	
030F	783	SYS status reg save	
0310-0312	784-785	USR function jump	(B248)
0314-0315	788-789	Hardware interrupt vector	(EA31)
0316-0317	790-791	Break interrupt vector	(FE66)
0318-0319	792-793	NMI interrupt vector	(FE47)
031A-031B	794-795	OPEN vector	(F34A)
031C-031D	796-797	CLOSE vector	(F291)
031E-031F	798-799	Set-input vector	(F20E)
0320-0321	800-801	Set-output vector	(F250)
0322-0323	802-803	Restore I/O vector	(F333)
0324-0325	804-805	INPUT vector	(F157)
0326-0327	806-807	Output vector	(F1CA)
0328-0329	808-809	Test-STOP vector	(F6ED)
032A-032B	810-811	GET vector	(F13E)
032C-032D	812-813	Abort I/O vector	(F32F)
032E-032F	814-815	Warm start vector	(FE66)
0330-0331	816-817	LOAD link	(F4A5)
0332-0333	818-819	SAVE link	(F5ED)
033C-03FB	828-1019	Cassette buffer	
0340-037E	832-894	(Sprite 13)	
0380-03BE	896-958	(Sprite 14)	
03C0-03FE	960-1022	(Sprite 15)	
0400-07FF	1024-2047	Screen memory	
0800-9FFF	2048-40959	Basic ROM memory	
8000-9FFF	32768-40959	Alternate: ROM plug-in area	
A000-BFFF	40960-49151	ROM: Basic	
A000-BFFF	49060-59151	Alternate: RAM	
C000-CFFF	49152-53247	RAM memory, including alternate	
D000-D02E	53248-53294	Video Chip (6566)	
D400-D41C	54272-54300	Sound Chip (6581 SID)	
D800-DBFF	55296-56319	Color nybble memory	
DC00-DC0F	56320-56335	Interface chip 1, IRQ (6526 CIA)	
DD00-DD0F	56576-56591	Interface chip 2, NMI (6526 CIA)	
D000-DFFF	53248-53294	Alternate: Character set	
E000-FFFF	57344-65535	ROM: Operating System	
E000-FFFF	57344-65535	Alternate: RAM	
FF81-FFF5	65409-65525	Jump Table, Including:	

FFC6 - Set Input channel [handwritten: 65478]
FFC9 - Set Output channel [handwritten: 65481]
FFCC - Restore default I/O channels [handwritten: 65484]
FFCF - INPUT [handwritten: 65487]
FFD2 - PRINT [handwritten: 65490]
FFE1 - Test Stop key [handwritten: 65505]
FFE4 - GET [handwritten: 65508]

	Address	Description
49060	A000;	ROM control vectors
49072	A00C;	Keyword action vectors
49142	A052;	Function vectors
49188	A080;	Operator vectors
49218	A09E;	Keywords
49414	A19E;	Error messages
	A328;	Error message vectors
	A365;	Misc messages
	A38A;	Scan stack for FOR/GOSUB
	A3B8;	Move memory
	A3FB;	Check stack depth
	A408;	Cneck memory space
	A435;	°out of memory°
	A437;	Error routine
	A469;	BREAK entry
	A474;	°ready.°
	A480;	Ready for Basic
	A49C;	Handle new line
	A533;	Re-chain lines
	A560;	Receive input line
	A579;	Crunch tokens
	A613;	Find Basic line
	A642;	Perform [NEW]
	A65E;	Perform [CLR]
	A68E;	Back up text pointer
	A69C;	Perform [LIST] 42652
	A742;	Perform [FOR]
	A7ED;	Execute statement
	A81D;	Perform [RESTORE]
	A82C;	Break
	A82F;	Perform [STOP]
	A831;	Perform [END]
	A857;	Perform [CONT]
	A871;	Perform [RUN]
	A883;	Perform [GOSUB]
	A8A0;	Perform [GOTO]
	A8D2;	Perform [RETURN]
	A8F8;	Perform [DATA]
	A906;	Scan for next statement
	A928;	Perform [IF]
	A93B;	Perform [REM]
	A94B;	Perform [ON]
	A96B;	Get fixed point number
	A9A5;	Perform [LET]
	AA80;	Perform [PRINT#]
	AA86;	Perform [CMD]
	AAA0;	Perform [PRINT]
	AB1E;	Print string from (y.a)
	AB3B;	Print format character
	AB4D;	Bad input routine
	AB7B;	Perform [GET]
	ABA5;	Perform [INPUT#]
	ABBF;	Perform [INPUT]
	ABF9;	Prompt & input
	AC06;	Perform [READ]
	ACFC;	Input error messages

(continuará)

Depósito Legal Sep. B. 2133-1958 - Altés, s.l., Caballero 87, Barcelona



Microprocesador: 6502 de MOS TECHNOLOGY de 8 bits.
Memoria: 5 Kbytes de RAM ampliables a 32 K 20 Kbytes de ROM ampliables a 28 K
Pantalla: 23 lineas de 22 caracteres Modulador para conectar a un televisor normal. Salida para monitor de video.
Colores: 8 para el marco, 16 para el fondo de la pantalla y ocho para los caracteres individuales, video inverso.
Gráficos: Semi-gráficos por teclado y alta resolución por redefinición del generador de caracteres (situándolo en RAM). Definición de 176 por 184 puntos.
Teclado: Tipo QWERTY de 62 teclas más cuatro de función definibles por el usuario.
Sonido: Tres voces de tres octavas cada una decaladas una octava entre sí, resultando una extensión total de cinco octavas. Un generador de ruido aleatorio afinable para efectos especiales, un control general de volumen.
Programación: Lenguaje BASIC, intérprete residente en ROM de 8 K. Posibilidad de interceptar las funciones del Basic para crear nuevas instrucciones «a medida». El Basic del Vic es uno de los más rápidos actualmente en el mercado.
Complementos: Port de usuario de 8 bits entrada/salida más dos señales de sincronismo.
Bus de expansión para ampliaciones de memoria y periféricos.
Port de juegos con conexión para dos potenciómetros (paddles), y una palanca de juegos (joystick).
Almacenamiento de masa: Unidad de cassette C2N de diseño especial para registrar programas y datos (ficheros secuenciales).

VIC-1540 UNIDAD DE DISCO

Capacidad total: 174848 bytes por disco.
Secuencial: 168656 bytes por disco.
Entradas de directorio: 144 por disco.
Sectores por pista: De 17 a 21.
Bytes por sector: 256.
Pistas: 35.
Bloques: 683 (644 bloques libres).
Soportes de información: Discos estandar de 5 1/4 pulgadas, de una sola cara y densidad simple.
Sistema operativo: DOS de COMMODORE inteligente (tiene procesador propio y no ocupa memoria del ordenador central).

VIC-1515 IMPRESORA

Método de impresión: Matriz de 5×7 puntos, impacto por un solo martillo.
Modo caracteres: Mayúsculas y minúsculas, símbolos, números y caracteres gráficos del VIC-20.
Modo gráfico: Puntos direccionables (bit image). Siete puntos verticales por columna, 480 columna máximo.
Velocidad: 30 caracteres/segundo, de izquierda a derecha, unidireccional.
Caracteres/Linea: Máximo 80. (Posibilidad de impresión en doble ancho).
Espaciado entre lineas: 6 lineas/pulgada -modo caracteres, 9 lineas/pulgadas - modo gráfico.
Velocidad de salto de lineas: 5 saltos/seg. -modo caracteres, 7,5 saltos/seg. - modo gráfico.
Alimentación de papel: Arrastre por tractor.
Ancho de papel: Entre 4,5 y 8 pulgadas.
Copias: Original más dos copias.

CARTUCHOS

Ayuda programador: Este cartucho facilita la edición y depuración de programas en Basic. Instrucciones y comandos: RENUMBER, MERGE, FIND, CHANGE, DELETE, AUTO, TRACE, STEP, OFF, KEY, EDIT, PROG, DUMP, HELP y KILL.

Super expander: Intercepta el Basic del VIC permitiendo incrementar sus instrucciones y comandos en aplicaciones gráficas, de sonido y juegos. Instrucciones y comandos: KEY, GRAPHIC, COLOR, POINT, REGION, DRAW, CIRCLE, PAINT, CHAR, SCNCLR, SOUND, RGR, RCOLR, RDOT, RPOT, RPEN, RJOY y RSND.

Monitor de lenguaje máquina: Este monitor altamente sofisticado facilita enormemente la depuración de programas en lenguaje máquina, es ideal como complemento del Basic para redactar y poner en marcha rutinas de alta velocidad y manejo de datos en tiempo real. Instrucciones y comandos: ASSEMBLE, BREAKPOINT, DISASSEMBLE, ENABLE VIRTUAL ZERO PAGE, FILL MEMORY, GO, HUNT, INTERPRET, JUMP TO SUBROUTINE, LOAD, MEMORY, NUMBER, QUICK TRACE, REGISTERS, REMOVE BREACPOINTS, SAVE, TRANSFER, WALK y EXIT TO BASIC.

Además existen cartuchos de ampliación de memoria de 3, 8 y 16 Kbytes.

CURSO DE INTRODUCCION AL BASIC PARTE I:

En forma de libro se ha editado la primera parte de un curso de Basic que parte «de cero» y está basado en el VIC-20. Va acompañado de dos cassettes con programas y ejercicios para autocontrol de los progresos en el aprendizaje.

MODULO DE EXPANSION DE MEMORIA:

Acabado en metal de gran robustez, permite la conexión de un máximo de 6 cartuchos simultáneamente, aloja al VIC y al modulador de vídeo y permite colocar encima el televisor, tiene alojamiento para accesorios y asegura una óptima conexión del VIC a sus periféricos.







microelectrónica y control, s. a.

Taquígrafo Serra, 7 5.° Telf. 250 51 03. BARCELONA-29
Princesa, 47 3.° G. Telf. 248 95 70. MADRID-8